ИСП Институт Системного Программирования
Российской Академии наук

ISSN 2079-8156 (Print)
ISSN 2220-6426 (Online)

Труды Института Системного Программирования РАН

Proceedings of the Institute for System Programming of the RAS

Том 26, выпуск 5

Volume 26, issue 5

Москва 2014

ИСП **Российская Академия наук**
Институт Системного Программирования

Труды Института Системного Программирования

Том 26
выпуск 5

Под редакцией
академика РАН В.П. Иванникова

Москва 2014

УДК004.45

Труды Института системного программирования: Том 26, выпуск 5.
/Под ред. Академика РАН В.П. Иванникова/ – М.: ИСП РАН, 2014.

В этом выпуске Трудов Института системного программирования РАН публикуются статьи, написанные по материалам докладов, которые были представлены на пятой ежегодной международной конференции "Облачные вычисления. Образование. Исследования. Разработка".

ISSN 2079-8156 (Print)

Содержание

П р е д и с л о в и е

В этом выпуске Трудов Института системного программирования РАН публикуются статьи, написанные по материалам докладов, которые были представлены на пятой ежегодной международной конференции "Облачные вычисления. Образование. Исследования. Разработка". Конференция прошла 4-5 декабря 2014 г в здании Президиума российской академии наук.

В 2014 г. конференция была посвящена различным аспектам технологий параллельных и распределенных вычислений, в том числе высокопроизводительным вычислениям в облаках, безопасности распределенных систем, хранению и анализу сверхбольших массивов данных, предметно-ориентированным, виртуальным web-лабораториям. В рамках конференции также был проведен тематический семинар по применению современных высокопроизводительных технологий и свободного программного обеспечения (пакет OpenFOAM) для решения задач механики сплошной среды.

Конференция проводится ИСП РАН при поддержке Российской академии наук и компаний-партнеров (HP, Dell, NVIDIA и д.р.), совместно с которыми реализуется ряд программ в области параллельных и распределенных вычислений: "Университетский кластер" (http://www.unicluster.ru), открытая лаборатория по технологиям больших данных (BigDataOpenLab - http://www.bigdataopenlab.ru), исследовательский центр CUDA (CUDA Research Center). Информационным партнером конференции является издательский дом "Открытые системы".

Доктор физико-математических наук А.И. Аветисян

Стохастическая модель процесса идентификации сервисов информационной системы

Г.Н. Циперман <g.tsiperman@voskhod.ru>
ФГУП НИИ «Восход», 119607, Россия, г. Москва,
ул. Удальцова, дом 85.

Аннотация. В статье рассматривается задача оценки объема трудозатрат на проектирование функциональных требований к информационной системе в сервис-ориентированной архитектуре. Для этого предлагается стохастическая модель процесса идентификации сервисов информационной системы, позволяющая при минимальных исходных данных дать оценку ожидаемого количества объектов проекта и связей между ними. Модель основана на представлении процесса декомпозиции автоматизируемого бизнес-процесса как ветвящегося случайного надкритического процесса Гальтона-Ватсона. Проектирование связей между элементами декомпозиции моделируется как процесс построения связного случайного графа в модели Эрдеша-Реньи. Предсказания модели подтверждаются проверкой на экспериментальных данных.

Ключевые слова: сервис, идентификация, стохастическая модель, ветвящийся процесс, случайный граф, оценка времени проектирования, трудозатраты, сервис-ориентированная архитектура.

1. Введение

Процесс идентификации сервисов информационной системы (ИС) в парадигме сервис-ориентированной архитектуры (СОА)представляет собой суть проектирования ИС, включающего стадии разработки концепции, технического и рабочего проектирования.

На раннем этапе проектирования трудно предсказать сложность проекта, характеризуемую количеством и структурой связей между фактами предметной области, которые порождают соответствующие требования к сервисам ИС. Однако, для практики проектирования ИС этот вопрос имеет особую актуальность, вызванную необходимостью обеспечения объективной стоимости проекта.

Поэтапная оценка такой сложности не актуальна, т.к. в сложившейся практике создания ИС отдельное финансирование стадий проекта, когда стоимость каждой последующей стадии оценивается на основе результатов предыдущей, является исключительно редким явлением. Заказчик хочет получить оценку стоимости проекта до того, как проект начнется.

Данная работа посвящена изучению общих свойств процесса идентификации сервисов, обеспечивающих прогнозирование трудоемкости стадий формирования функциональных требований к создаваемой ИС.

2. Модель процесса идентификации сервисов

Основным методом идентификации сервисов является декомпозиция автоматизируемого бизнес-процесса, в результате которойсначала определяется дерево бизнес-функций [1]. Каждая бизнес-функция представляет собой описание некоторого промежуточного результата, получаемого при выполнениисновного бизнес-процесса.

Например, бизнес-процесс проектирования технического сложного объекта (ТСО) на первом уровне декомпозируется на следующие бизнес-функции:

- Инициация проектирования. Результатом является организация проекта по проектированию ТСО.
- Анализ структуры проектируемого ТСО, в результате которого формируются функциональная и логическая модель изделия.
- Разработка конструкции, предполагающая получение конструктивной модели ТСО и определение его комплектующих.
- Многоаспектный анализ предлагаемой конструкции ТСО, определяющий на основе технических и экономических расчетов соответствие характеристик проекта требованиям технического задания.
- Завершение проектирования, результатом которого является проект ТСО.

В свою очередь, каждая бизнес-функция может быть декомпозирована далее, исходя из более детальных промежуточных результатов ее выполнения.

Понятно, что для составления полной картины бизнес-процесса на каждом уровне декомпозиции должна быть определена последовательность выполнения бизнес-функций, т.е. объекты декомпозиции должны быть связаны управляющими потоками. Такая модель называется сценарием бизнес-функции и представляет собой связный граф, узлами которого являются объекты декомпозиции, а ребрами – потоки управления.

Конечный уровень декомпозиции бизнес-процесса соответствует описанию необходимых действий (операций), которые должен выполнить субъект

бизнес-процесса, чтобы получить требуемый промежуточный результат. Объекты этого уровня называются бизнес-операциями.

Таким образом, декомпозиция бизнес-функции бизнес-процесса включает в себя как бизнес-функции следующего уровня, так и бизнес-операции и описывается соответствующим сценарием.

Собственно, идентификация сервисов ИС начинается с определения сервисов операций. Под сервисом операции мы будем понимать спецификацию функций информационной системы, поддерживающих исполнение соответствующей бизнес-операции. Между бизнес-операцией и сервисом операции устанавливается однозначное соответствие. В связи с этим мы можем рассматривать их в связке как единый конструкт. Визуальное представление процесса декомпозиции бизнес-процесса приведено нарис. 1.

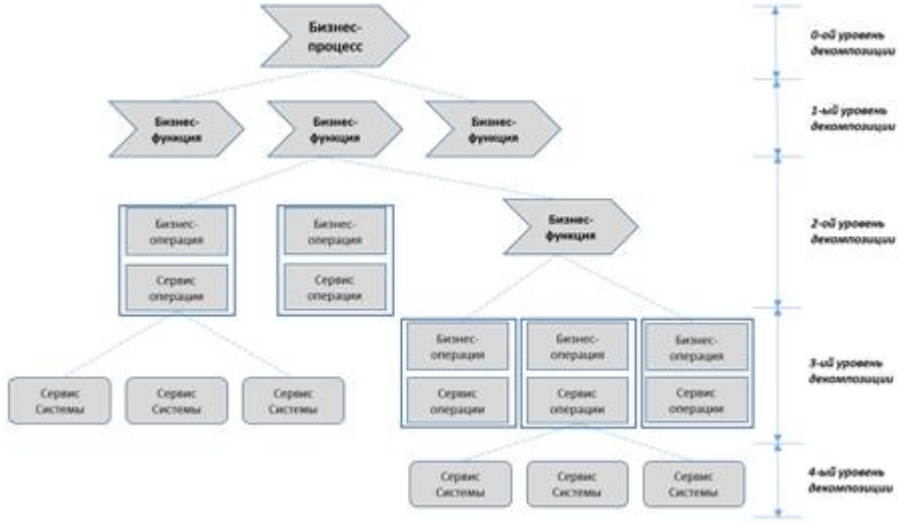


Рис. 1. Дерево декомпозиции бизнес-процесса.

Дадим формальное определение процессу декомпозиции бизнес-процесса.

Пусть $B_{fnc} = \{f_i\}$ и $B_{opr} = \{o_j\}$, где $i, j \in \mathbb{N}$– множества бизнес-функций и бизнес-операций бизнес-процесса (вместе с сервисами операций). Определим отношение декомпозиции:

$$D^b : B_{fnc} \to B_{fnc} \cup B_{opr} \quad (1)$$

$$D_i^b = D^b(f_i), f_i \notin D_i^b \quad (2)$$

При этом

$$\left\{ \begin{array}{c} \forall \begin{pmatrix} D_i^b = (f_{i1}, \dots, f_{ik}, o_{i1}, \dots, o_{ip}), \\ D_j^b = (f_{j1}, \dots, f_{jl}, o_{j1}, \dots, o_{js}) \end{pmatrix} \in D^b, \forall i, j, p, s, n \in \mathbb{N}: \\ D_i^b \cap D_j^b = \emptyset \\ D_1^b \cup D_2^b \cup \dots \cup D_n^b = (B_{fnc} \setminus f_0) \cup B_{opr} \end{array} \right\} \quad (3)$$

В (3)f_0– исходный бизнес-процесс. Таким образом, каждая бизнес-функция и бизнес-операция присутствуют только в одной декомпозиции.

Определяем условие завершения декомпозиции, отражающее тот факт, что декомпозиция имеет смысл, если в результате порождается не менее одного элемента:

$$\forall i \in \mathbb{N}: \left| D_i^b \right| \geq 1 \quad (4)$$

Дадим формальное определение сценарию бизнес-функции. Рассмотрим множество пар элементов

$$S^b = \{(s_i, s_j); \forall i, j \in \mathbb{N}, i \neq j;\ s_{i,j} \in (B_{fnc} \setminus f_0) \cup B_{opr}\} \quad (5)$$

Сценарием бизнес-функции f_k с декомпозицией D_k^b называется подмножество

$$S_k^b \subset S^b$$

$$S_k^b = \{(s_i, s_j); D_k^b \subset \{s_{i,j}\}; \forall i, j \in \mathbb{N}, i \neq j, s_{i,j} \in (B_{fnc} \setminus f_0) \cup B_{opr}\} \quad (6)$$

Сценарий бизнес-функции в отличие от ее декомпозиции может включать бизнес-функции и бизнес-операции из других декомпозиций. Этот факт отражает повторное использование конструктов при проектировании информационной системы. Как видно из (6), в случае, если бизнес-функция не имеет собственной декомпозиции она может иметь сценарий, в который входят элементы иных декомпозиций.

Пара элементов сценария соответствует отношению передачи управления между элементами и, вообще говоря, является ориентированной. Однако, нас интересует оценка количества таких пар и вопросы ориентации этой связи мы рассматривать не будем.

Вдентификация сервисов ИС представляет собой декомпозицию сервисов операций: сервисы системы обеспечивают выполнение функциональности, специфицированной сервисами операций. Аналогично последовательность вызова сервисов системы моделируется управляющими потоками, связывающими сервисы в соответствующий сценарий сервиса операции. Формально это определяется следующим образом.

Пусть $C_{opr} = \{c_i\}$множество сервисов операций. Связь сервисов операций модели с бизнес-операциями обеспечивается однозначным отношением зависимости

$$L: o_i \to c_i, o_i \in B_{opr}, c_i \in C_{opr} \quad (7)$$

Как отмечалось выше, пару (o_i, c_i)в силу характера однозначного отношения между ними мы будем рассматривать как единый объект.

Пусть $C_{sys} = \{r_i\}$ множество системных сервисов модели. Аналогично (1) - (3) определяем отношение декомпозиции для сервисов:

$$D^s: C_{opr} \to C_{sys} \quad (8)$$

$$D_i^s = D^s(r_i) \quad (9)$$

$$\begin{cases} \forall \left(D_i^s = (r_{i1}, \dots, r_{ik}), D_j^s = \left(r_{j1}, \dots, r_{jl}\right)\right) \in D^s, \forall i, j, k, l, n \in \mathbb{N}: \\ D_i^s \cap D_j^s = \emptyset \\ D_1^s \cup D_2^s \cup \dots \cup D_n^s = C_{sys} \\ |D_i^s| \geq 1 \end{cases} \quad (10)$$

Сервис операции, не имеющий самостоятельной декомпозиции, тем не менее, может иметь сценарий, включающий элементы декомпозиции других сервисов операций. Сценарий сервиса операции определяется аналогично сценарию бизнес-функции (5) – (6):

Рассматриваем множество пар элементов

$$S^s = \left\{\left(r_i, r_j\right); i \neq j;\ r_{i,j} \in C_{sys}\right\} \quad (11)$$

Сценарием сервиса операции c_k с декомпозицией D_k^s называется подмножество

$$S_k^s \subset S^s$$

$$S_k^s = \left\{\left(r_i, r_j\right); D_k^s \subset \left\{r_{i,j}\right\}; i \neq j, r_{i,j} \in C_{sys}\right\} \quad (12)$$

2.1. Вероятностная модель процесса декомпозиции

Процесс декомпозиции модели информационной системы мы будем рассматривать как случайный ветвящийся процесс Гальтона-Ватсона[2]. Обобщим оператор декомпозиции:

$$D(x_i) = \begin{cases} D^b(x_i), x_i = f_i \in B_{fnc} \\ D^s(x_i), x_i = r_i \in C_{sys} \end{cases} \quad (13)$$

Множество элементов декомпозиции будем обозначать как

$$D_i = D(x_i) = (x_{i1}, x_{i2}, \dots, x_{in}) \quad (14)$$

В таких обозначениях ветвящийся процесс может быть описан следующим образом:

$$D_0 = D(0) = (f_0)$$
$$D_1 = D(f_0) = (x_1, x_2, \dots, x_n)$$
$$D_{21} = D(x_1) = (x_{11}, \dots, x_{1m}); \dots; D_{2n} = D(x_n) = (x_{n1}, \dots, x_{nl}) \quad (15)$$
$$D_{\underbrace{ki\dots j}_{k}} = D\left(x_{\underbrace{i\dots j}_{k-1}}\right) = \left(x_{\underbrace{i\dots j1}_{k}}, \dots, x_{\underbrace{i\dots jp}_{k}}\right)$$

Пусть $\xi_{ki\dots j}$ – количество элементов декомпозиции $D_{ki\dots j}$. Элементарной реализацией процесса декомпозиции назовем набор

$$\omega = \left(\xi_0, \xi_{1i}, \xi_{2ip}, \dots, \xi_{kip\dots j}\right) \quad (16)$$

Вероятностное пространство$(\Omega, \mathcal{F}, P)$процесса декомпозиции определяется на пространстве событий, представляющем собой множество всех элементарных реализаций

$$\Omega = \{\omega\} \quad (17)$$

При этом, σ-алгебра$\mathcal{F}$ порождается подмножествами пространства Ω, а вероятностная мера P задается соотношениями:

$$P\{\xi_{kip\dots j} = k\} = p_k$$
$$P\{(\xi_0, \xi_{1i}, \xi_{2ip}, \dots, \xi_{kip\dots j}) = (k_0, k_{1i}, k_{2ip}, \dots, k_{kip\dots j})\} \quad (18)$$
$$= p_{k_0} p_{k_{1i}} p_{k_{2ip}} \dots p_{k_{kip\dots j}}$$

Процесс декомпозиции, как и соответствующий ему процесс Гальтона-Ватсона, можно представить в виде плоского дерева, корнем которого является бизнес-процесс f_0. Будем называть k-ым поколением процесса декомпозиции или уровнем дерева подмножество декомпозиций процесса

$$I_k = \{D_{ki\dots j}\} \quad (19)$$

Количество элементов k-ого поколения будем обозначать

$$Z(k) = |\{D_{ki\dots j}\}| = \sum_{i\dots j} \xi_{ki\dots j} \quad (20)$$

Сформулируем основную задачу исследования этапа определения структурных элементов информационной системы. Для процесса декомпозиции, представляющего основной механизм определения структуры, $Z(0) = 1$ и, положим $Z(1) = N_0$, гдеN_0неслучайная величина, соответствующая количеству бизнес-функций первого уровня декомпозиции бизнес-процесса, специфицированных при предварительном изучении темы проекта создания ИС.

Будем рассматривать процесс декомпозиции $\widetilde{D}_0$ как N_0 подпроцессов $\widetilde{D}_{1i}$, представляющих собой надкритические процессы Гальтона-Ватсона с ненулевой вероятностью вырождения α (соответственно, вероятность невырождения процесса γ = 1 - α). Надкритичность процесса означает, что математическое ожидание числа элементов декомпозиции

$$E[\xi] > 1 \tag{21}$$

Требуется найти наиболее вероятное число элементов декомпозиции (ожидаемое число структурных компонент модели ИС) при указанных условиях.

2.2. Вероятностная модель построения сценария

Положим мы определили структуру модели информационной системы, спроектировав множество ее компонент, представляющее собой множество декомпозиций

$$D = \{D_i\}, |D| = n \tag{22}$$

На следующем этапе проектирования мы должны построить сценарии бизнес-функций и сервисов операций, определяющих использование ИС в автоматизируемом бизнес-процессе. Мы будем рассматривать структуру системы перед началом построения сценариев как несвязный граф *G(n)*, состоящий из *n* изолированных вершин. При таком подходе, исходя из смысла проектирования информационной системы, результатом проектирования сценариев должен явиться сценарий использования системы, представляющий собой связный граф *G(n,M)*, где *M* – количество ребер этого графа. Формально, используя определения (6) и (12):

$$M = |S| = \sum_k |S_k|$$

$$S = \{S_k\} = \tag{23}$$

$$= \begin{cases} S_k^b = \{(s_i, s_j); D_k^b \subset \{s_{i,j}\}; \forall i,j \in \mathbb{N}, i \neq j, s_{i,j} \in (B_{fnc} \setminus f_0) \cup B_{opr}\} \\ S_k^s = \{(r_i, r_j); D_k^s \subset \{r_{i,j}\}; i \neq j, r_{i,j} \in C_{sys}\} \end{cases}$$

Задачей исследования процесса построения сценария использования проектируемой информационной системы является определение необходимого количества времени для получения сценария S или, иными словами, формирования связного графа G(n,M).

Будем рассматривать граф G(n,M) как случайный граф в модели Эрдеша-Реньи. Формальное описание этой модели приведем, следуя изложению из монографии [3].

Определим вероятностное пространство всех графов $\mathfrak{L}(n,M)$порядка n и размера M, каждый элемент которого равновероятен. Очевидно

$$0 \le M \le N = \binom{n}{2}, |\mathfrak{L}(n,M)| = \binom{N}{M} \quad (24)$$

Тогда вероятность формирования любого графа *G(n,M)* из этого пространства

$$P\{G = G(n,M)\} = \binom{N}{M}^{-1} \quad (25)$$

В общем случае M функционально зависит от n: M=M(n).

Введем понятие процесса на случайном графе.

Пусть $V = \{1,2,\dots,n\}$. Процессом $\tilde{G} = (G_t)_0^N$, где $N = \binom{n}{2}$ называется марковский процесс на V такой, что:

1) $(G_0)_0^N = 0$
2) Каждый G_tпредставляет собой граф на V
3) G_t имеет t ребер, где $t = 1,2,\dots,N$
4) $G_0 \subset G_1 \subset \cdots$

Пусть $\widetilde{\mathfrak{L}} = \{\tilde{G}\}, |\widetilde{\mathfrak{L}}| = N!$вероятностное пространство, элементами которого являются равновероятные процессы$\tilde{G}$.Отображение$\widetilde{\mathfrak{L}} \longrightarrow \mathfrak{L}(n,M)$определяется как

$$\tilde{G} = (G_t)_0^N \longrightarrow G(n,M) \quad (26)$$

Иначе: состояние процесса G_tв момент времени $t = M$ отождествляется с элементом пространства $\mathfrak{L}(n,M)$.

Мы рассмотрим процесс формирования сценария использования информационной системы как процесс на случайном графе. Момент завершения формирования сценария соответствует моменту T_S, когда граф *G(n,M)* становится связным. Под связностью мы будем понимать вершинную односвязность $(k(G) = 1)$. На практике это означает, что полностью определена связь бизнес-функций и входящих в них бизнес-операций и все состояния проектируемой информационной системы связаны между собой хотя бы одним маршрутом.

3. Метрики процесса идентификации сервисов

В качестве основной метрики процесса идентификации сервисов ИС, отражающей его стохастические закономерности, мы рассмотрим общее

количество элементов функциональной модели бизнес-процесса: бизнес-функций, бизнес-операций и системных сервисов

$$T_D = |D| = \sum_{i=0}^{k} Z(i) \quad (27)$$

Определение процесса декомпозиции (15) служит основой для рассмотрения всех элементов в единой вероятностной модели, несмотря на различие их природы. Другие метрики процесса (общее время проектирования и его эффективность) определяются количеством элементов модели.

Модель процесса проектирования в части определения структуры функциональной модели автоматизируемого бизнес-процесса определяется следующими исходными предположениями:

1) В основе определения структурных элементов функциональной модели лежит процесс декомпозиции автоматизируемого бизнес-процесса.
2) Процесс декомпозиции рассматривается как ветвящийся процесс Гальтона-Ватсона с граничными условиями $Z(0) = 1;\ Z(1) = N_0$, где N_0– неслучайная величина.
3) Рассматриваемый ветвящийся процесс является надкритическим с ненулевой вероятностью вырождения. Процесс вырождается в случае, если каждая декомпозиция k-ого поколения включает 0 или 1 элемент.
4) Число элементов декомпозиций, относящихся к k-ому поколению имеет распределение Пуассона

$$P\{\xi_k = x\} = \frac{\lambda^x}{x!} e^{-\lambda} \quad (28)$$

5) Математическое ожидание распределения элементов декомпозиций $\lambda = E[\xi]$не зависит от поколения процесса.

Другой метрикой, зависящей от общего количества элементов функциональной модели, является количество связей между элементами. Это количество мы будем определять,как минимально необходимое для связности графа функциональной модели бизнес-процесса.

3.1. Ожидаемое количество элементов функциональной модели

Для количества элементов k-ого поколения ветвящегося процесса математическое ожидание определяется как [4]

$$Z(i) = \lambda^i, i = (0,1,2, \dots) \quad (29)$$

Соответственно оценка для общего количества элементов модели, учитывая независимость λ от номера поколения, может быть определено по формуле суммы геометрической прогрессии

$$n = \sum_{i=0}^{k} Z(i) = \frac{\lambda^{k+1} - 1}{\lambda - 1} \quad (30)$$

Учитывая граничные условия рассматриваемого процесса декомпозиции, мы будем рассматривать общий процесс как N_0 независимых процессов, начинающихся со второго уровня модели (считаем с нулевого уровня). Тогда выражение (30) примет вид

$$n = 1 + N_0 \left(\frac{\lambda^k - 1}{\lambda - 1} \right) \quad (31)$$

Для того, чтобы сделать оценку (31) применимой необходимо оценить математическое ожидание элементов единичной декомпозиции и ожидаемое число уровней модели.

Обсудим вероятности вырождения ветвящегося процесса α и, соответственно, невырождения $\gamma = 1 - \alpha$. Интуитивно понятно, что чем выше вероятность невырождения процесса, тем больше структурных элементов декомпозиции удастся получить. В практике проектирования информационных систем это соответствует глубине понимания проектировщиком автоматизируемого бизнес-процесса: чем больше деталей предметной области известно, тем больше фактов (элементов) может быть отражено в функциональной модели бизнес-процесса. В связи с этим мы будем называть вероятность γ детальностью модели бизнес-процесса, обратную вероятность α – ее неопределенностью.

Детальность проектируемой модели может служить входным параметром для оценки основных метрик процесса проектирования. Чем выше детальность, тем подробнее должна быть модель и, соответственно, больше трудозатрат такая работа потребует.

Известным фактом теории ветвящихся процессов является зависимость вероятности вырождения надкритического ветвящегося процесса от математического ожидания λ [4]. Неопределенность модели является наименьшим неотрицательным корнем уравнения

$$f(\alpha) = \alpha \tag{32}$$

Здесь $f(\alpha)$вероятностная производящая функция, задающая распределение случайной величины ξ. Для пуассоновского распределения [5]

$$f(\alpha) = \sum_{i=0}^{\infty} P\{\xi = i\}\alpha^i = \sum_{i=0}^{\infty} \frac{(\lambda\alpha)^i}{i!} e^{-\lambda} = e^{\lambda(\alpha-1)} = e^{-\lambda\gamma} \tag{33}$$

Подставив (33) в (32), легко видеть

$$\lambda = -\frac{ln(1-\gamma)}{\gamma} = -\frac{ln(\alpha)}{1-\alpha} \tag{34}$$

Уравнение (34) определяет зависимость ожидаемого количества элементов единичной декомпозиции от детальности (неопределенности) функциональной модели бизнес-процесса.

Для оценки ожидаемого количества уровней декомпозиции воспользуемся следующими соображениями. Пусть вырождение процесса декомпозиции начинается с уровняK. Мы рассмотрим каждый элемент этого уровняx_{Ki} как начальный для отдельного процесса $\widetilde{D}_{Ki}$. В этом случае для каждого такого процесса декомпозиции общее количество элементов определяется выражением [4]

$$n_{Ki} = \frac{1}{1 - f'(\alpha)} \tag{35}$$

Учитывая (32) и (33) выражение (35) принимает вид

$$n_{Ki} = \frac{1}{1-\lambda\alpha} \tag{36}$$

Как видно из (36) число элементов, возникающих при вырождении любого из процессов $\widetilde{D}_{ki}$,определяется выражением

$$\Delta n_{deg} = \frac{1}{1-\lambda\alpha} - 1 = \frac{\lambda\alpha}{1-\lambda\alpha} \tag{37}$$

Диапазон, в котором лежит Δn_{Ki}определяется неравенством[1]

$$\underline{\Delta n_{deg}} = \left\lfloor \frac{\lambda\alpha}{1-\lambda\alpha} \right\rfloor \le \Delta n_{deg} \le \left\lceil \frac{\lambda\alpha}{1-\lambda\alpha} \right\rceil = \overline{\Delta n_{deg}} \tag{38}$$

Общее число процессов $\widetilde{D}_{Ki}$равно λ^{K-1}– количеству элементов наK-ом уровне декомпозиции бизнес-процесса, относящихся к подпроцессу $\widetilde{D}_{1i}$. Тогда

[1]Мы будем обозначать символами $\lfloor a \rfloor$ и $\lceil a \rceil$ значения округления числа aвниз и вверх соответственно.

количество элементов, входящих в следующие за K-ым уровни (уровни вырождения), будет определяться как $\Delta N_{deg} = \lambda^{K-1}\,\Delta n_{deg}$ и лежит в диапазоне

$$\underline{\Delta N_{deg}} = \lambda^{K-1}\,\underline{\Delta n_{deg}} \le \Delta N_{deg} \le \overline{\Delta N_{deg}} = \lambda^{K-1}\overline{\Delta n_{deg}} \qquad (39)$$

Значение количества элементов декомпозиции на уровне вырождения процесса $\widetilde{D}_{1i}$, определяемое выражением (39), мы будем относить к последнему K+1-ому поколению декомпозиции этого процесса. Для того, чтобы оценить значение K, положим, что уровень K является последним, т.е. на уровне вырождения нет элементов. Это означает, что

$$\Delta N_{Ki} = \lambda^{K-1}\frac{\lambda\alpha}{1-\lambda\alpha} < 1 \qquad (40)$$

Разрешая неравенство (40)относительно K, получаем оценку

$$K = \left\lceil \frac{ln(1-\lambda\alpha) - ln(\lambda\alpha)}{ln(\lambda)} \right\rceil + 1 \qquad (41)$$

На рис. 2 приведен график зависимости количества уровней декомпозиции Kфункциональной модели бизнес-процесса от детальности λ.

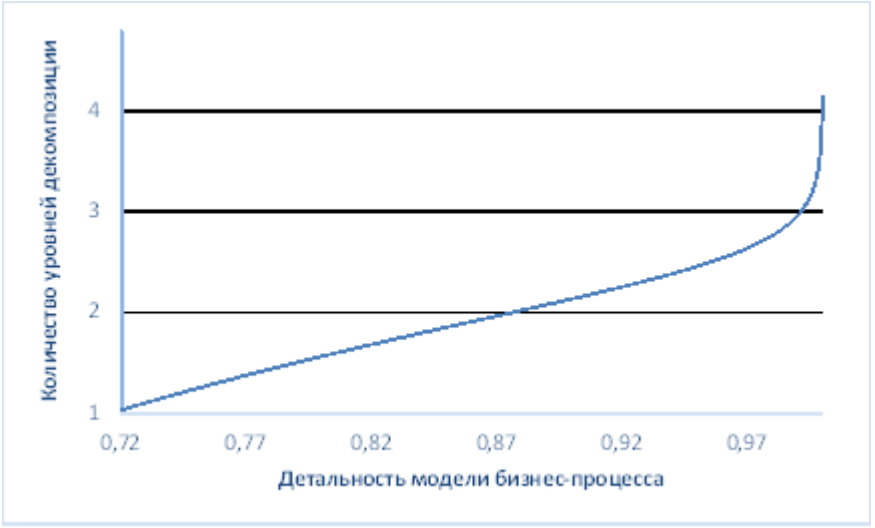


Рис. 2. График зависимости количества уровней декомпозиции функциональной модели бизнес-процесса от детальности.

Определение уровня начала вырождения декомпозиции основывалось на предположении, что следующий уровень модели, соответствующий вырождению, не содержит элементов декомпозиции. На самом деле, это не так: мы должны учесть элементы на уровне вырождения (на практике их вполне заметное количество). Однако, полученные значения уровня начала декомпозиции в предположении, с ростом числа элементов уровня вырождения меняются медленно. Можно показать, что K растет медленнее логарифма числа элементов на уровне вырождения декомпозиции. В связи с этим оценка, определяемая уравнением (41), является вполне приемлемой и, что важно, согласуется с реальной практикой проектирования, где в основном декомпозиция заканчивается на третьем уровне, а уровень 5 встречается достаточно редко.

Теперь мы имеем все необходимые данные для оценки общего числа элементов декомпозиции исходного бизнес-процесса:

$$T_D = n + N_0 \Delta N_{deg} = 1 + N_0 \left(\frac{\lambda^K - 1}{\lambda - 1} + \Delta N_{deg} \right) \tag{42}$$

Учитывая неравенство (39) ожидаемое количество элементов декомпозиции определяется диапазоном

$$\begin{aligned} \underline{T_D} = \left\lfloor 1 + N_0 \left(\frac{\lambda^K - 1}{\lambda - 1} + \underline{\Delta N_{deg}} \right) \right\rfloor \le T_D \le \overline{T_D} = \\ = \left\lceil 1 + N_0 \left(\frac{\lambda^K - 1}{\lambda - 1} + \overline{\Delta N_{deg}} \right) \right\rceil \end{aligned} \tag{43}$$

3.2. Время проектирования сценариев

Как отмечалось, момент завершения проектирования сценариев модели определяется наступлением в случайном процессе $\tilde{G}$ связности графа *G(n,M)*, вершинами которого являются элементы декомпозиции бизнес-процесса (бизнес-функции, бизнес-операции и системные сервисы). При этом T_Sопределяется количеством ребер M в модели.

В работе [3] показано, что для $\forall k \in \mathbb{N}, x \in \mathbb{R}$, если

$$M(n) = \frac{n}{2}\left(ln(n) + kln\left(ln(n)\right) + x + o(1) \right) \tag{44}$$

то вероятность того, что состояние процесса $\tilde{G}$ является k-связным определяется предельным соотношением

$$P\left\{k\left(G(n,M)\right) = k\right\} \longrightarrow 1 - e^{-e^{-x/k!}} \tag{45}$$

Используем это соотношение следующим образом. Положим, что требуемая вероятность связности сценариев определяется детальностью модели, т.е.

$$P\{k(G(n,M)) = 1\} = \gamma \tag{46}$$

Используя требование (45), из (46)получаем

$$x = -ln(-ln(1-\gamma)) \tag{47}$$

Тогда, с учетом (47), время T_S, необходимое для того, чтобы граф *G(n,M)* стал связным, на основании выражения (44) можно оценить как

$$T_S = M(T_D) = \frac{T_D}{2}\left(ln(T_D) + ln(ln(T_D)) - ln(-ln(1-\gamma))\right) \tag{48}$$

Здесь T_Dопределяется соотношением (43). Диапазон для T_Sсоответственно определяется как

$$M\left(\underline{T_D}\right) \le T_S \le M\left(\overline{T_D}\right) \tag{49}$$

3.3. Оценка трудоемкости проектирования

На основании оценки количества элементов декомпозиции модели бизнес-процесса мы можем оценить ожидаемуютрудоемкость проектирования. Исходим из того, что трудоемкость, необходимая для декомпозиции пропорционально количеству элементов декомпозиции (43), а трудоемкость построения сценариев бизнес-функций и сервисов операций пропорциональна количеству ребер связного графа модели, определяемому выражением (48):

$$T_P = k_d T_D + k_s T_S \tag{50}$$

Здесь k_d и k_s– экспертно определяемые коэффициенты. Эти коэффициенты определяются временем, необходимым для проектирования одним человеком одного элемента декомпозиции и одной связи сценария соответственно. Длярасчетов мы будем исходить из следующей экспертной оценки:

$$\begin{aligned} k_d &= 1{,}5 \div 2{,}0 \text{ чел} \cdot \text{час} \\ k_s &= 0{,}1 \div 0{,}3 \text{ чел} \cdot \text{час} \end{aligned} \tag{51}$$

Отметим, что эти экспертные оценки отражают опыт автора и, вообще говоря, зависят от квалификации специалиста. В каждом конкретном случае представленные коэффициенты должны уточняться с учетом специфики проектной организации.

В качестве примера прогнозную оценкутрудоемкости проектирования в зависимости от детальности модели бизнес-процесса представляют табл. 1 для $N_0 = 4$ ирис.3 для $N_0 = 8$.

Табл. 1. Оценка трудоемкости проектирования для $\boldsymbol{N_0 = 4}$**.**

γ	K	T_D		T_S		Трудоемкость (чел*час)	
		min	max	min	max	min	max
0,83	2	14	23	21	43	34	59
0,86	2	15	24	23	44	37	61
0,87	2	15	24	22	43	37	61
0,88	3	38	62	79	149	100	169
0,9	3	42	68	89	164	111	185
0,93	3	50	82	107	201	132	224
0,96	3	64	109	141	276	170	301
0,98	3	85	149	194	391	228	415
0,985	3	95	168	220	447	256	470
0,99	3	111	197	263	534	301	554
0,991	3	115	205	273	558	312	577
0,992	4	581	1042	1929	3810	1741	3227
0,993	4	624	1123	2089	4139	1875	3488

Рис 3. График зависимости трудоемкости от детальности модели при $\boldsymbol{N_0 = 8}$.

Скачки на графике зависимости времени проектирования от детальности модели соответствуют изменению уровня начала вырождения модели. Интерпретация таких переходов связана со стадиями проектирования информационной системы. Уровень начала вырождения 2 соответствует стадии разработки концепции системы и предполагает определение автоматизируемых операций. Третий или четвертый уровень соответствует стадии технического проектирования, на котором специфицируются требования к системным сервисам. Наконец, пятый и последующие уровни предполагают детальную разработку требований к программным компонентам системных сервисов и соответствуют стадии рабочего проектирования.

4. Проверка модели на экспериментальный данных

Для проверки предложенной стохастической модели процесса идентификации сервисов информационной системы мы рассмотрим два выполненных автором технических проекта информационных систем. Первый проект связан с проектированием аналитической (OLAP) системы, а второй касается информационной системы поддержки концептуального проектирования технически сложных объектов морской техники (проект ТСО МТ).

Проверка будет осуществлена по следующим направлениям:

1) Проверка распределения количества элементов декомпозиции каждого уровня на соответствие распределению Пуассона[2]
2) Попадание общего количества элементов декомпозиции проектной модели ξ_f в рамки теоретически предсказанного интервала
3) Соответствие общего количества связей в сценариях проектной модели M_fпредсказаниям стохастической модели процесса проектирования.

4.1. Проект OLAP-системы

Распределение количества элементов декомпозиции ξ_(ki…j) в модели OLAP-системы представляет рис. 4.

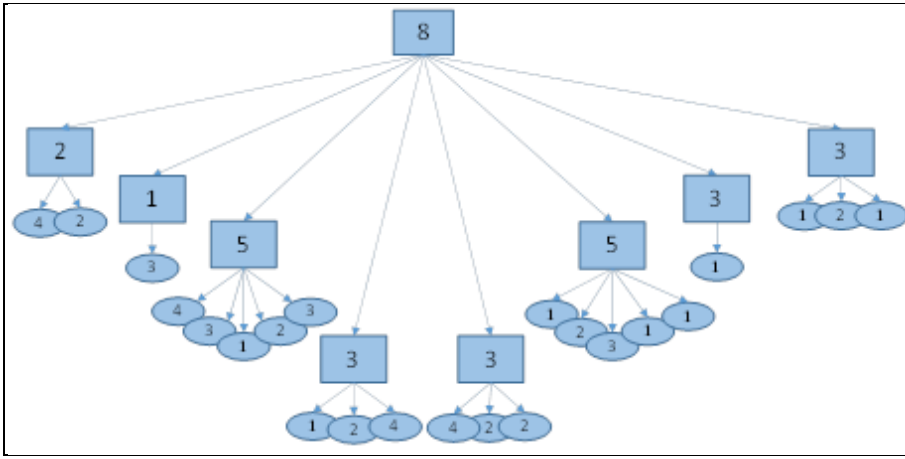


Рис. 4. Структура декомпозиции модели OLAP-системы.

Исходные характеристики модели:

- Количество элементов декомпозиции первого уровня $N_0 = 8$
- Количество уровней декомпозиции *k*=3 (начиная с нулевого)
- Количество элементов декомпозиции $\xi_f = 83$

[2] Проверка гипотезы о том, что распределение количества элементов в декомпозиции соответствует распределению Пуассона, выполнено при помощи интернет-ресурса http://math.semestr.ru/group/poisson-examples.php.

- Количество связей в сценариях$M_f = 249$.

Распределение количества элементов декомпозиции для уровней 2 и 3, а также для всей модели, полученное на основе анализа структуры модели, представляет табл. 2.

Табл. 2. Распределение количества элементов декомпозиции в модели OLAP-системы.

i	ξ_{2i}	ξ_{3i}	ξ_t
1	1	8	9
2	1	7	8
3	4	4	8
4	0	4	4
5	2	0	2
$E[\xi]$	3,13	2,17	2,42

Проверка гипотезы о соответствии распределения количества элементов декомпозиции распределению Пуассона дает следующие результаты:

1) Для уровня 2. При уровне значимости 0,4 критическое значение статистики Пирсона $K_{kp} = 7{,}38$, а наблюдаемое значение $K_{\text{набл}} = 0{,}87$.
2) Для уровня 3. При уровне значимости 0,4 критическое значение статистики Пирсона $K_{kp} = 7{,}38$, а наблюдаемое значение $K_{\text{набл}} = 3{,}72$.
3) Для всей модели. При уровне значимости 0,4 критическое значение статистики Пирсона $K_{kp} = 9{,}35$, а наблюдаемое значение $K_{\text{набл}} = 3{,}91$.

Во всех случаях наблюдаемое значение статистики Пирсона не попадает в критическую область ($K_{\text{набл}} < K_{kp}$), поэтому справедливо предположение о том, что данные выборки имеют распределение Пуассона.
Расчет определяет для $\lambda = \xi_t = 2{,}42$ ожидаемые значения:

$$\gamma = 0{,}881$$

$$K = 3$$

$$76 \le T_D \le 122$$

$$192 \le T_S \le 343$$

Как видно, экспериментальное значение ξ_fблизко к нижней границе диапазона. Это объясняется тем, что уровень вырождения в модели отсутствует, т.е. последний уровень декомпозиции равен теоретическому уровню начала вырождения K=3.

4.2. Проект ТСО МТ

На рис. 5 представлено распределение количества элементов декомпозиции $\xi_{ki\ldots j}$в модели информационной системы поддержки концептуального проектирования ТСО МТ.

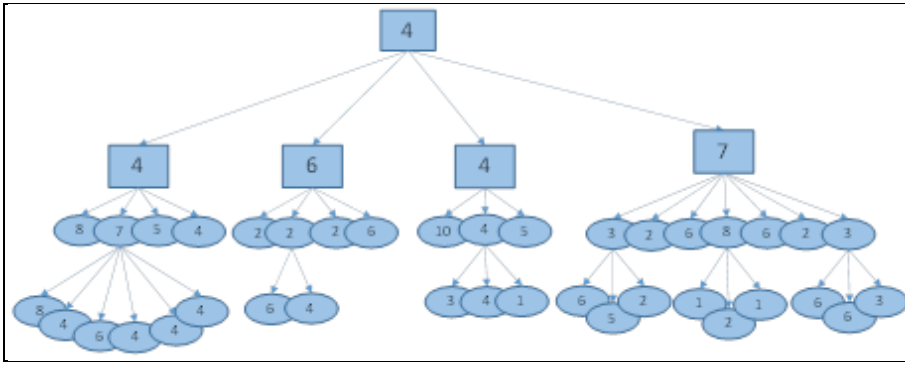


Рис. 5. Структура декомпозиции модели информационной системы поддержки концептуального проектирования ТСО МТ.

Исходные характеристики модели:

- Количество элементов декомпозиции первого уровня $N_0 = 4$
- Количество уровней декомпозиции $k = 4$ (начиная с нулевого)
- Количество элементов декомпозиции $\xi_f = 187$
- Количество связей в сценариях $M_f = 345$.

Распределение количества элементов декомпозиции для уровней 2 и 3, а также для всей модели, полученное на основе анализа структуры модели, представляет табл. 3.

Табл. 3. Распределение количества элементов декомпозиции в модели проекта ТСО МТ

i	ξ_{2i}	ξ_{3i}	ξ_{3i}	ξ_t
1	0	5	3	7
2	0	2	2	4
3	2	2	2	10
4	0	2	6	3
5	1	3	1	9
6	1	1	5	2
7	0	2	0	3
8	0	0	1	0
9	0	1	0	1
$\boldsymbol{E}[\boldsymbol{\xi}]$	5,25	4,72	4,00	4,43

Проверка гипотезы о соответствии распределения количества элементов декомпозиции распределению Пуассона дает следующие результаты:

1) Для уровня 2. При уровне значимости 0,05 критическое значение статистики Пирсона $K_{kp} = 3{,}84$, а наблюдаемое значение $K_{\text{набл}} = 2{,}63$.
2) Для уровня 3. При уровне значимости 0,05 критическое значение статистики Пирсона $K_{kp} = 12{,}59$, а наблюдаемое значение $K_{\text{набл}} = 9{,}01$.
3) Для уровня 4. При уровне значимости 0,05 критическое значение статистики Пирсона $K_{kp} = 11{,}07$, а наблюдаемое значение $K_{\text{набл}} = 6{,}68$.
4) Для всей модели. При уровне значимости 0,05 критическое значение статистики Пирсона $K_{kp} = 14{,}07$, а наблюдаемое значение $K_{\text{набл}} = 9{,}21$.

Во всех случаях наблюдаемое значение статистики Пирсона не попадает в критическую область ($K_{\text{набл}} < K_{kp}$), поэтому справедливо предположение о том, что данные выборки имеют распределение Пуассона.

Расчет определяет для $\lambda = \xi_t = 4{,}43$ ожидаемые значения:

$$\gamma = 0{,}987$$

$$K = 3$$

$$102 \leq T_D \leq 181$$

$$239 \leq T_S \leq 487$$

Экспериментальное значение $\xi_f \approx \overline{T_D}$близко к верхней границе диапазона (расхождение составляет 3%). Близость к верхней границе ожидаемого диапазона объясняется тем, что в модели присутствует уровень вырождения, т.е. последний уровень декомпозиции на единицу превосходит теоретический уровень начала вырождения K=3.

Таким образом, можно сделать заключение, что предлагаемая стохастическая модель процесса идентификации сервисов информационной системы дает хорошее приближение к экспериментальным данным.

Список литературы

[1]. Г.Н. Циперман. Применение метода адаптивной кластеризации для проектирования сложных информационных систем. III Научно-практическая конференция «Актуальные проблемы системной и программной инженерии». Сборник научных трудов. Москва, МЭСИ, 6 -7 июня 2013 г.

[2]. В.А. Ватутин, А.М. Зубков. Ветвящиеся процессы. I. Сборник «Итоги науки и техники». Серия «Теория вероятностей. Математическая статистика. Теоретическая кибернетика». Москва, ВИНИТИ, 1985, том 23, стр. 3-67.

[3]. BelaBollobas. RandomGraphs. Secondedition, CambridgeUniversityPress 2001

[4]. В.А. Ватутин. Ветвящиеся процессы Беллмана-Харриса. Лекционные курсы НОЦ. Математический институт им. В.А. Стеклова. Москва, МИАН, 2009.

[5]. Н. Алон, Дж. Спенсер. Вероятностный метод. Москва, БИНОМ. Лаборатория знаний, 2011.

The stochastic model of information system services identification process

G. Tsiperman <g.tsiperman@voskhod.ru>
Federal State Unitary Enterprise Organization "VOSKHOD", 85 Udaltsova Str., Moscow, 119607, Russian Federation

Abstract. This article refers to labor costs estimation of information system functional requirements design problem in a service-oriented architecture. Stochastic model of information system services identification process is offered for this purpose, which allows providing the estimation of expected project objects and interrelations amount using minimal initial data. This model is based on automated business-process decomposition representation as a branching random supercritical Galton-Watson process. Decomposition elements relation design is been modeled as connected random graph construction process by Erdösh-Rényi model. Model predictions have been confirmed through verification by experimental data.

Keywords: service, identification, stochastic model, branching process, random graph, design time estimation, labor costs, service-oriented architecture

References

[1]. G.N.Tsiperman. Primenenie metoda adaptivnoi klasterizatsii dly aproekturovaniya slozhnyh informatsionnyh system [Application of a method of adaptive clustering for complex information systems design] III Nauchno-prakticheskaya konferentsiya «Aktualnye problemy sistemnoii programmnoi inzhenerii» [III Scientific-practical conference "Actual problems of system and software engineering"]. Sbornik nauchnyh trudov [Collection of proceedings]. Moscow, MESI, 6 -7 June 2013 r. (in Russian).

[2]. V.A. Vatutin, A.M. Zubkov. Vetvyaschiesya protsessy. I [Branchingprocess. I]. Sbornik «Itogi nauki I tekhniki». [The book "The Results of science and technology"]. Moscow, VINITI, 1985, vol. 23, 3-67 p.(in Russian).

[3]. BelaBollobas. RandomGraphs. Secondedition, CambridgeUniversityPress 2001

[4]. V.A. Vatutin. Vetvyaschiesya protsessy Bellman-Harris [Branching processes Bellman-Harris]. LektsionnyekursyNOTS [Lecture courses]. Mathematical instituteV.A. Steklov. Moscow, MIAN, 2009.(in Russian).

[5]. N. Alon, J.H. Spencer. The Probabilistic Method. Third Edition. JohnWiley&Sons, 2008.

Оценка сложности крупноблочных облачных вычислений, использующих арифметику повышенной точности

С.С.Толстых<inf@tstu.ru>
В.Е.Подольский <director@director.tixmcnit.tambov.su>
Тамбовский государственный технический университет

Аннотация. В статье рассмотрены вопросы оценки сложности крупноблочных облачных вычислений с повышенной точностью. Данная разработка направлена на решение в облаке задач математического моделирования с особыми требованиями точности. В частности речь идет о получении прецизионно-доверительного решения задач со сложными связями между подзадачами в виде крупных блоков и временем счета, значительно превышающим время передачи информации между ними. Предлагается методология оценки сложности задач подобного рода, используемая для построения оптимальных по производительности вычислительных систем, функционирующих в облачной среде.

Ключевые слова: крупноблочные параллельные вычисления, облачные вычисления, прецизионные вычисления, прецизионно-доверительное решение задач, вещественная арифметика повышенной точности.

1. Введение

Работа направлена на облачно-ориентированное решение обширного класса задач, основанных на математических моделях со сложной топологией: дополнительно он характеризуется большим объемом вычислений с плавающей точкой при повышенных требованиях к достоверности результата. К таким задачам относится, например, задача поиска неполной системы сочетаний мероприятий по рекреации водоемов промышленного региона, задача о проектировании системы воздухообмена в замкнутых ограниченных объемах сложной конфигурации и многие другие. Задачи подобного рода требуют ответственного решения, т.к. от полученных результатов могут зависеть объемы капиталовложений, безопасность жизнедеятельности, категорируемое качество изделий. Похожие задачи рассматривались в 80-е в работах, посвященных декомпозиции сложных химико-технологических схем. Фактически, выигрыш по времени счета на однопроцессорных ЭВМ достигался за счет поиска оптимального итерируемого множества (ОИМ), снижения размерности общей системы нелинейных уравнений; удавалось

найти такие разрезы орграфа технологической схемы, что суммарное количество переменных, по которым осуществлялся итерационный расчет математической модели, было минимальным (например, химического цеха с числом аппаратов порядка 100). Задача оптимальной декомпозиции решалась методом ветвей и границ, путем упорядоченного перебора дуг орграфа, предполагаемых к разрыву, на матрице контуров, и включения переменных, соответствующих дуге, в ОИМ минимальной мощности. После нахождения ОИМ глобальные итерации производились методом Ньютона-Раффсона или его модификациями: вместо огромной системы нелинейных уравнений (СНУ), удавалось свести задачу к СНУ, вполне пригодной к однократному решению. В 80-е мощностей отечественных ЕС ЭВМ не хватало, чтобы на основе математической модели химического цеха находить оптимальные технологические параметры, в таких случаях даже в сокращенном варианте СНУ решались очень медленно.

В настоящее время появилась возможность решать в облаке задачи глобальной оптимизации не только в детерминированном случае, но и в условиях неопределенности, неизбежно возникающих при недостаточной наблюдаемости объекта исследования. Объем вычислений в таких задачах резко возрастает по сравнению с детерминированными постановками, а решение необходимо находить в многомерной области с размерностью, превышающей изначальную размерность детерминированной задачи. Как правило, получение решение задач оптимизации в условиях неопределенности ранее ограничивалось теми редкими случаями, когда достаточно было однократного решения общей задачи в условиях ее локализации до одного процесса или аппарата. Кроме того, решение подобных задач в условиях сильной нелинейности ограничений получить путем построения аналитических выпуклых оболочек практически невозможно, что увеличивает и без того высокую вычислительную сложность. Если, пусть даже в условиях аналитически полученных выпуклых оболочек, перейти на уровень системы объектов (цеха, например), необходимость высокопроизводительного распараллеливания очевидна, так же как и то, что отдельные элементы вычислительной системы являются крупноблочными.

Следует отметить, что при повышении размерности крупноблочных задач математического моделирования теряется представление о точности получаемого решения. Можно ли доверять полученному решению? Если, например, в расчетах используется многостадийные математические модели кинетики химических реакций органического синтеза, скорости реакций могут отличаться на порядки, и система дифференциальных уравнений становится жесткой. К подобным задачам сводятся также математические модели взрывных процессов, процессов горения и обжига.

Практика вычислений в рамках пакетов численных методов дает основания полагать, что даже при решении тестовых задач, связанных с жесткими системами, точности формата представления чисел с плавающей точкой при

16-ти десятичных разрядов в мантиссе не хватает. Требуется привлечение классов и/или функций для работы с арифметикой повышенной точности. При этом время счета (особенно с участием стандартных математических функций типа exp(x)) резко возрастает. Ранее легко решаемая, задача даже небольшой размерности превращается в вычислительную проблему.

Далеко не всегда удается использовать уже ставшие традиционными методы распараллеливания, разработанные для структурно простых задач (например, ряда задач математической физики), с весьма значительным количеством однотипных элементов (конечные элементы, конечные разности). С другой стороны, в случае перехода на уровень, когда отдельный аппарат становится элементом сколь-нибудь значительной системы, возникает тенденция рассматривать структурно простой элемент как требующий использования мощной вычислительной площадки (кластера), таким образом, снова вырисовывается крупный вычислительный блок, а вся система в целом становится крупноблочной.

Характер организации параллельных вычислений в новых условиях требует, прежде всего, новых теоретических обоснований и методик крупноблочного распараллеливания в облаке, с учетом требований получения прецизионно-доверительного решения и при возможном наличии в вычислительной системе глобальных итерационных циклов. Мотивом создания новой теории является необходимость оценки вычислительной сложности, как критерия минимизации: в условиях непростых топологий вычислений важно правильно распределить задачи по ресурсам, добиваясь значительного сокращения общего времени счета.

Предлагаемая методология оценки сложности высокопроизводительных крупноблочных вычислений нацелена на построение архитектуры решения задачи в облаке, чьи ресурсы используются для прецизионно-доверительных решений сложных задач математического моделирования и оптимизации с вышеуказанными особенностями. В основе этой методологии представление архитектуры облака в виде взвешенного ориентированного графа с дугами, отождествляемыми с укрупненными за счет агрегирования блоками. Вес этих дуг сопоставляется с теоретической сложностью численных методов, т.к. агрегирование всего набора вычислительных подзадач подразумевает, что один блок решает одну типичную задачу вычислительной математики. При этом в формулу входят еще и такие параметры, как размер мантиссы, размерность задачи, требуемая точность решения (если в локальном методе присутствуют итерационные циклы типа while(…), например, итерационные методы решения СНУ). Вершины орграфа отождествляются при этом с серверами облака, раздающими исходные данные и инициирующими расчеты в кластерах или отдельных компьютерах (если используется, например, вузовская компьютерная сеть).

В статье используется терминология и обозначения, принятые в монографии [1]. Работы выполняются в соответствии с Проектом № 1346 из реестра

государственных заданий высшим учебным заведениям и научным организациям в сфере научной деятельности.

2. Общетеоретические положения

Считаем, что структура задачи нам известна из результатов структурной идентификации исследуемой системы $S^{(0)}$ в соответствии с целями исследования (фактически $S^{(0)}$ – исходная система, например, цех по производству ацетилена; природо-промышленная система региона и др.) и представляет собой орграф $G^{(0)} = \left(V^{(0)}, D^{(0)}, \Gamma^{(0)}\right)$, где $V^{(0)}$ – набор вершин, $D^{(0)}$ – набор дуг, $\Gamma^{(0)}$ – весовые характеристики дуг. При этом $n^{(0)} = \left|V^{(0)}\right|$ – число вершин, $m^{(0)} = \left|D^{(0)}\right| = \left|\Gamma^{(0)}\right|$ – число дуг орграфа структурно-сложной вычислительной задачи. Необходимо найти такой орграф облачной вычислительной системы (ОВС) для решения исходной задачи, чтобы итоговая вычислительная сложность была минимальной:

$$\left(G^* \equiv \left(V^*, D^*, \Gamma^*\right)\right) = Arg \min_{G} \theta\left(G\left(G^{(0)}\right)\right), \quad (1)$$

где θ – оценка вычислительной сложности орграфа $G = G\left(G^{(0)}\right)$. Функция $G\left(G^{(0)}\right)$ характеризует процесс осуществления структурной идентификации на этапе синтеза ОВС.

Каждая из дуг искомого орграфа G отождествляет собой расчет, характеризуемый следующими величинами:

1. Вычислительной сложностью $\theta\left(d_k\right)$;
2. Числом разрядов в мантиссе при осуществлении арифметических операций $T\left(d_k\right)$;
3. Числом входных переменных, которые участвуют в организации итерационного процесса, при условии, что эта дуга разрывается $I\left(d_k\right)$.

Следует заметить, что в простейшем случае – без агрегации вычислений – параметричность $\gamma_k, k = \overline{1,m}$ является функционалом вида $\gamma_k = \gamma_k\left(\theta\left(d_k\right), T\left(d_k\right), I\left(d_k\right)\right)$, в частности предлагается следующий вид

$$\gamma_k = I(d_k) \times \frac{\bar{E}_{\theta,k} + \theta(d_k)}{\theta(d_k)} \times \frac{\bar{E}_{T,k} + \left[T(d_k)\right]^{\iota_k}}{T(d_k)}, \qquad (2)$$

где

1. $\bar{E}_{\theta,k} \geq 1$ – параметр, определяемый экспертами, и являющийся мерой верификативной адекватности математического описания и моделируемого объекта исходной системы $S^{(0)}$; так, например, феноменологическая модель обычно в большей мере готова к верификации (в частном случае к идентификации), нежели модель, построенная по принципу «черного ящика» и основанная на аппроксимации экспериментальных данных; в параметр $\bar{E}_{\theta,k}$ можно включать, например, такие показатели, как полноту учета факторов и соответствие изучаемому явлению параметров математической модели; таким образом параметр $\bar{E}_{\theta,k}$ выражает предпочтительность математического описания и, если он равен 1, роль вычислительной сложности дуги d_k в ее результатной параметричности минимальна.

2. $\bar{E}_{T,k} \geq 1$ – параметр, определяющий рост параметричности γ_k при увеличении количества разрядов мантиссы; естественно предположить, что, если например, мы хотим сравнить параметричность двух дуг, в одной из которых число разрядов равно 100, а в другой – 50, последняя более предпочтительна для итерирования; с другой стороны значительное влияние на выбор дуги к разрыву может сыграть количество разрываемых при этом контуров – чем оно больше, тем больше информационная нагрузка этой дуги; предлагается оценивать данный параметр как контурность дуги d_k (число контуров орграфа, в которых участвует дуга);

3. $\iota_k \geq 1$ – показатель, характеризующий чувствительность времени счета к росту числа разрядов при расчетах, связанных с дугой d_k; определяется как степень чувствительности величины $\theta(d_k)$ к росту требований точности получаемого локального решения в дуге d_k; например, если при расчетах дуги используется метод Гаусса для решения СЛАУ размерности *n*, и, в свою очередь, чем больше размерность *n*, тем на принципиальном уровне точнее расчет d_k (такого рода задачи могут встречаться, например, при конечно-разностном подходе к численному решению дифференциальных уравнений в частных производных), то $\iota_k \sim 3$, т.к. с ростом размерности СЛАУ вычислительная сложность данного метода пропорциональна кубу размерности СЛАУ.

2.1. Основные обозначения и термины

Будем полагать, что орграф G является результатом структурной идентификации облачной вычислительной системы (ОВС) S, являющейся объектом исследования. Структура системы стационарна, таким образом, процесс ее выявления – структуризация $\mathrm{Str}(S)$ – является отображением, ставящим в соответствие системе S взвешенный орграф G

$$\mathrm{Str}(S):\ S \longrightarrow G. \tag{3}$$

Оценка сложности $\theta(S)$ является интегральной количественной характеристикой общей вычислительной нагрузки ОВС, показателем, позволяющим произвести упорядочение различных предлагаемых извне структур ОВС и выбрать вариант, имеющий при прочих равных условиях минимальную сложность.

Функционирование системы S подчиняется целям ее существования и это выражается автоморфизмом, отражающим цель вычислений

$$Aim(S):\ S \longrightarrow S. \tag{4}$$

Орграф G – та среда, в рамках которой решаются задачи $T(G) = (T_1(G), ...)$, согласованные с $Aim(S)$.

Решение задач из кортежа $T(G)$ производится на шкале $\theta(G)$ после замены оценок сложности системы S оценками сложности орграфа

$$\theta(G) = des\,\theta(S), \tag{5}$$

где «$= des(\circ)$» – характеризация, операнд слева является характеристикой операнда справа [1].

Орграф G представляет собой кортеж $G = (V, D, \Gamma)$. В составе кортежа G находятся

- $V = (v_i, i = 1..n)$ – кортеж вершин;
- $D = \left(d_k = des\left(v_i \to v_j\right), k = 1..m; i, j \in \overline{\{1, n\}}, i \neq j\right)$ – кортеж дуг;

– $\Gamma = \left(\gamma_k, k = 1..m; \gamma_k :=: d_k\right)$ – кортеж параметричностей; функтор $\gamma\left(d_k\right)$ переопределяет γ_k и является процедурной моделью процесса параметризации дуг; $\gamma_k :=: d_k$ означает «γ_k соответствует d_k»[1].

Далее по тексту в графических иллюстрациях и некоторых формулах вершины "v_i" могут обозначаться алиасами (дополнительными именами) "i".

Орграф ОВС G имеет следующие особенности:

1) параметричности – суть вещественные скаляры, превышающие единицу: $\gamma_k \geq 1$. Таким образом, единица выступает в роли точки отсчета параметричностей. Дуге $d_k = des\left(v_i \to v_j\right)$ сопоставлен вес $\gamma_k \in R^{\geq 1}, k = 1..m$, где $R^{\geq 1}$ – множество положительных действительных чисел с точкой отсчета 1.

2) ограничимся рассмотрением орграфов без изолированных вершин и кратных дуг, и это ограничение соответствует выполнению условия

$$\left(\nexists v_i \in V : \nexists j \neq i : \left(v_i \to v_j \vee v_j \to v_i\right)\right) \wedge \nexists k_1, k_2 : d_{k_1} = d_{k_2}. \quad (6)$$

2.2. Ассоциативные представления элементов орграфа ОВС

Важно заранее оговорить, какие смысловые ассоциации возникают в представлении структуры ОВС: что именно ассоциируется с дугами, а что – с вершинами. В данном случае представляется наиболее удобным вкладывать основную вычислительную нагрузку в дуги, вершины при этом выступают в роли серверов, осуществляющих распределение информационно-вычислительных потоков. На рисунке 1 проиллюстрирована смысловая основа информационно-вычислительной нагрузки, приписываемой дуге орграфа.

[1] По определению этой операции [1] из того факта, что, к примеру, $a :=: b$ не следует $b :=: a$.

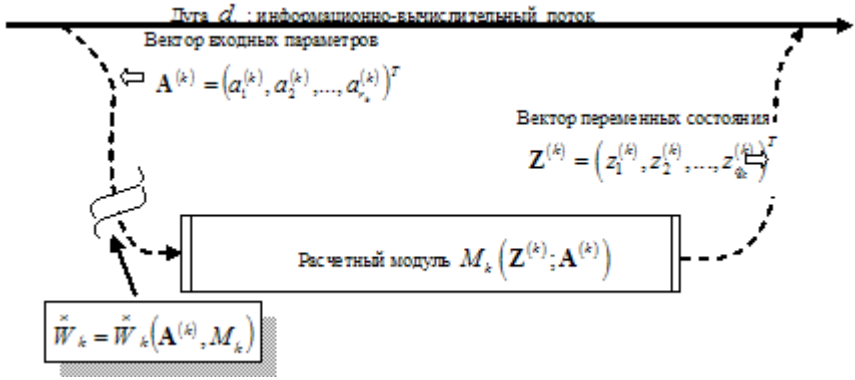


Рисунок 1. Вычислительная нагрузка дуги орграфа

Отметим следующее:

1) дуга d_k имеет два возможных состояния: обычное и разрываемое, в последнем информационно-вычислительная нагрузка содержит не только расчеты, но и функции управления итерационным циклом с последующим сигналом о завершенности/незавершенности итераций;

2) информационно-вычислительная нагрузка дуги в обычном состоянии определяется вычислительной сложностью расчетного модуля $M_k\left(\mathbf{Z}^{(k)}; \mathbf{A}^{(k)}\right)$, где $\mathbf{Z}^{(k)}$ – вектор переменных состояния, получаемый в результате вычислений, $\dim \mathbf{Z}^{(k)} = q_k$; $\mathbf{A}^{(k)}$ – вектор параметров, управляющих особенностями вычислений в расчетном модуле, $\dim \mathbf{A}^{(k)} = r_k$.

3) результатная информационно-вычислительная нагрузка дуги в условиях разрыва $\overset{\times}{d}_k :=: \overset{\times}{\gamma}_k$ может отличаться от обычного состояния $\overline{d}_k :=: \overline{\gamma}_k$; чем больше параметров входит в состав вектора $\mathbf{A}^{(k)}$, т.е. чем больше число r_k и тем выше нагрузка дуги.

4) выбор дуги для разрыва и последующая вслед за этим организация итерационного цикла производится на основе совокупной вычислительной сложности.

Под «совокупной вычислительной сложности понимается $\gamma_k = \circ\left(\overline{\gamma}_k, \overset{\times}{\gamma}_k\right)$, где

$\circ$ – функциональный абстрактор[2].

Сущность информационно-вычислительного потока представлена фреймом знаний, в его составе процедурные модели $M_k\left(\mathbf{Z}^{(k)}, \mathbf{A}^{(k)}\right)$ и $\gamma_k = \circ\left(\overline{\gamma}_k, \overset{\times}{\gamma}_k\right)$.

Информационная составляющая представлена в потоке векторами $\mathbf{Z}^{(k)}$ и $\mathbf{A}^{(k)}$, а вычислительная – вышеназванными процедурными моделями, и, возможно (если дуга разрывается) – процедурной моделью разрыва дуги[3] в виде предиката $\overset{\times}{W}_k = \overset{\times}{W}_k\left(\mathbf{A}^{(k)}, M_k\right) \in \{0,1\}$. Значение «0» соответствует продолжению итераций, а «1» - окончанию цикла. В случае, когда все $\overset{\times}{W}_k = 1,\ k = 1.\beta^*$, считаем, что вычислительная нагрузка в системе равна нулю (здесь β^* – общее число разрываемых дуг всей ОВС в целом).

2.3. Сопоставимость орграфов ОВС

Структурные свойства орграфа ОВС представлены совокупностью двух кортежей – кортежа вершин V и кортежа дуг D. Параметрические – кортежем Γ. Важным аспектом оценки сложности является построение шкалы орграфов, ранжирование. Структурные свойства орграфа при ранжировании имеют больший приоритет, нежели параметрические. Необходимо ввести соответствующие определения.

Определение 1.

Взвешенный орграф G сопоставим по структуре с не взвешенным орграфом $\underline{G}$, и это обозначается $G \approx \underline{G}$, если взвешенная матрица смежности взвешенного орграфа G и матрица смежности не взвешенного орграфа $\underline{G}$ равны с точностью до знака числа[4]

[2] В монографии [1] имеется строгое определение этого понятия применительно к категории структурной сложности абстрактных систем, но в данном случае более уместно воспользоваться новыми тенденциями развития объектно-ориентированного подхода в моделировании – «абстрактор – это объект класса, наследуемого от абстрактного класса контекстным образом».

[3] В частности, такой моделью может быть тот или иной итерационный метод решения систем уравнений.

[4] В формуле (7) и далее по тексту используется обозначение $\hookrightarrow$ – «в противном случае».

$$\left(\begin{Bmatrix} G = con(\mathbf{X}), \\ \underline{G} = con(\underline{\mathbf{X}}) \end{Bmatrix}, \begin{Bmatrix} x_{ij} = des(\gamma_k :=: (i \to j)), \\ \underline{x}_{ij} = des(\exists (i \to j) \Rightarrow 1 \subsetneq 0), \underline{x}_{ij} = sign(x_{ij}) \end{Bmatrix}; i, j = 1..n\right) = (1 :=: true). \quad (7)$$

Определение 2.

Взвешенные орграфы G_1 и G_2 называются идентичными по структуре, и это обозначается $G_1 \sim G_2$, если

$$\exists \underline{G}_1 : G_1 \approx \underline{G}_1, \exists \underline{G}_2 : G_2 \approx \underline{G}_2, \underline{G}_1 = \underline{G}_2 \Rightarrow G_1 \sim G_2 \quad (8)$$

Взвешенные орграфы могут иметь идентичную структуру, при этом параметричности могут находиться в отношении прямой пропорциональности.

Определение 3.

Орграфы G_1 и G_2, идентичные по структуре и имеющие сходство в том, что параметричности G_2 могут быть получены из параметричностей G_1 умножением на коэффициент α, называются сопоставимыми и это обозначается $G_1 \cong G_2$, если выполняется условие

$$\left(G_1 = con(\mathbf{X}_1), G_2 = con(\mathbf{X}_2); \left\{x_{1,ij} = \alpha x_{2,ij}\right\}; i, j = 1..n, i \neq j, \alpha > 0\right) = 1. \quad (9)$$

2.4. Аксиоматика интегральных оценок вычислительной сложности ОВС (начальный этап)

Для формализации оценки $\theta(G)$ необходим ряд аксиом сложности. Они служат теоретическим базисом вывода формул для количественной оценки сложности ОВС.

Аксиома 1 (об оценках сложности сопоставимых орграфов).

Если $G_1 \cong G_2$ и $1 \le \alpha \le \bar{\alpha}$, они сопоставимы по сложности, причем, если сложность $\theta(G_1)$ известна, величина $\theta(G_2)$ пропорциональна $\theta(G_1)$, а именно

$$\theta(G_2) = \upsilon(\alpha)\theta(G_1), \upsilon(\alpha) : \forall \alpha > \circ : \upsilon(\circ) < \upsilon(\alpha) < \upsilon(\bar{\alpha}), \bar{\alpha} > \alpha. \quad (10)$$

В формуле (10) функция $\upsilon(\alpha)$ выступает в роли коэффициента пропорциональности, причем на величину α в формулировке аксиомы

накладывается ограничение в виде двойного неравенства[5]. В более сложных случаях $\upsilon(\alpha)$ является полиномом степени, равной числу уровней иерархий в дереве разрывов, а величина $\bar{\alpha}$ – параметр структурной бифуркации (величина, чье плавное увеличение при определенном значении может вызвать резкое увеличение оценки сложности).

3. Метод лексиграфической нумерации сильно связных орграфов

Сложность системы, состоящей из ряда подсистем, не меньше сложности всей системы, в то же время, вполне очевидно, оценка сложности только тогда конструктивна, когда эта общеизвестная аксиома выполняется с точностью до знака «равенство» для изолированных подсистем. Поэтому, формализация оценок сложности должна быть согласована по способу применения с неделимостью системы на подсистемы, т.е. орграф системы при разработке оценок должен быть сильно связным.

*Гипотеза***.** Если бы в нашем распоряжении появился способ лексиграфической нумерации сильно связных орграфов, то следом возникла бы тенденция связать эту нумерацию с оценками сложности, т.е. тем самым произвести оцифровку шкалы сложности (см иллюстрацию на рисунке 2).

Для замены абстрактора $\circ(G)$ на конкретную функцию орграфу G ставится в соответствие инвариант $R(G)$ – уникальное целое число $R(G): \nexists G', G' \neq G \Rightarrow R(G) = R(G') \wedge \forall G' \subset G : R(G) > R(G')$. Уникальность состоит в том, что двух орграфов с одинаковым $R(G)$ не существует, и для любого подграфа его инвариант всегда строго меньше инварианта орграфа, которому он принадлежит. Фактически $R(G)$ – оценка структурной сложности орграфа по числу его дуг. Кроме известных аксиом сложности, в этой оценке учитывается непротиворечивое утверждение: «Если орграф G содержит больше дуг по сравнению с орграфом G', он сложнее».

Идея вычисления инварианта $R(G)$ состоит в накоплении битовых единиц, начиная с младших разрядов, по матрице смежности $\mathbf{X}$ с последующим преобразованием битовых строк в целое неотрицательное число

[5] Способ нахождения $\upsilon(\alpha)$ и $\bar{\alpha}$ предстоит найти в будущем. Это позволит резко увеличить эффективность вычислений оценок сложности: внутренняя вычислительная сложность расчета $\theta(G)$ – экспоненциальная по m.

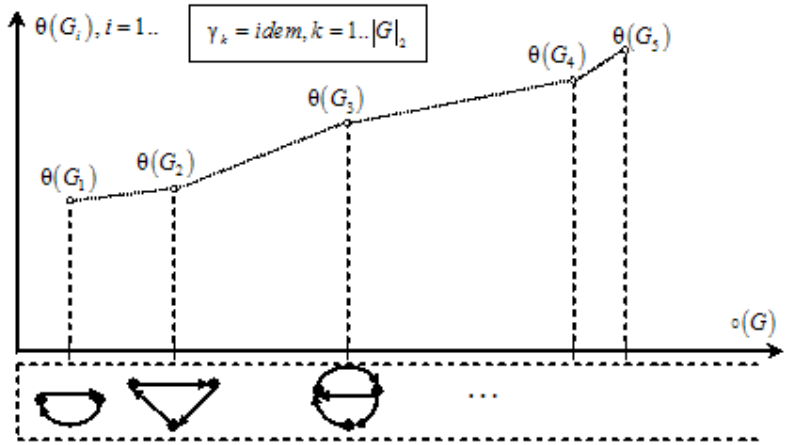


Рисунок 2. Иллюстрация к гипотезе о росте оценок сложности орграфов при их лексиграфическом перечислении в условиях постоянства параметричностей

$$
\begin{aligned}
&R(G) = R_0(G) + R_1(G), R_0(G) = \\
&0 + \left[\begin{matrix} n & n \\ \bullet & \bullet \\ i=1 & j=1 \end{matrix} \left(\left\{ \begin{matrix} i = j \Rightarrow "" \\ i \neq j \Rightarrow "".x_{ij} \end{matrix} \right\} \right) \right], R_1(G) = \sum_{i=1}^{|G|_2} 2^{i+k-1}, \\
&k = \left\lfloor \log_2 R_0\left(\widehat{G}\right) + \frac{1}{2} \right\rfloor, \widehat{G} = con\left(\widehat{\mathbf{X}}\right), G = con(\mathbf{X}), \\
&\dim \mathbf{X} = \dim \widehat{\mathbf{X}}, \widehat{x}_{ij} = 1, i = 1..n,\ j = 1..n, i \neq j, n = |G|_1
\end{aligned}
\tag{11}
$$

На рисунке 3 приведен пример, иллюстрирующий вычисление $R(G)$.

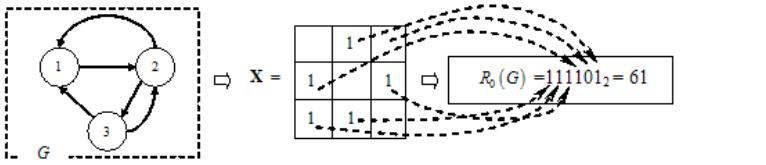


Рисунок 3. Пример вычисления инварианта орграфа

4. Структурная декомпозиция орграфа ОВС

В общем случае изначально орграф ОВС может содержать сильно связные компоненты (бикомпоненты), или, если их не было изначально, в процессе вычисления $\theta(G)$ орграф G рекурсивно упрощается, и появляются бикомпоненты. Таким образом, необходимо формализовать два кардинально разных состояния орграфа ОВС: 1) орграф G сильно связный; 2) орграф G не является сильно связным. В первом случае матрица достижимости

$$\mathbf{H} = \left(h_{ij}\right)_{n\times n}, h_{ij} = \begin{cases} \exists\left(i \to \ldots \to j\right) \vee i = j \Rightarrow 1, \\ \nexists\left(i \to \ldots \to j\right) \Rightarrow 0, i, j = 1..n \end{cases} \quad (12)$$

заполнена единицами полностью, во втором – лишь частично.

На рисунке 4 показаны три орграфа: 1) сильно связный; 2) дерево бикомпонент; 3) дерево вершин.

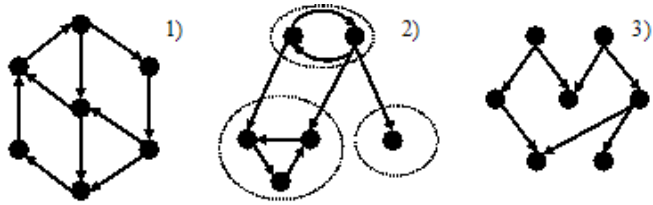


Рисунок 4. Три варианта орграфа (бикомпоненты – в овалах)

Рисунок 4 демонстрирует необходимость рассмотрения трех аспектов формализации $\theta(G)$:

1) оценка сложности деревьев вершин;
2) оценка сложности деревьев бикомпонент;
3) выбор упрощающих операций по приведению сильно связного орграфа к состоянию дерева;

5. Аксиоматика интегральных оценок вычислительной сложности ОВС: оценка сложности древовидных структур

Структура древовидных ОВС описывается в виде деревьев вычислений. Орграф G называется деревом[6], если в нем отсутствуют контуры и, чтобы

[6] В данном случае уместно уточнение – дерево вершин.

выделить эти орграфы в отдельный класс, вводим для них специальное обозначение $\hat{G}$.

Для оценки сложности деревьев сформированы следующие аксиомы:

Аксиома 2 (сложность элементарного дерева).

Оценка сложности элементарного дерева с двумя вершинами не меньше веса дуги, которая их соединяет

$$\left|\hat{G}\right|_1 = 2, \left|\hat{G}\right|_2 = 1 \Rightarrow \theta\left(\hat{G}\right) \geq \gamma_1, \tag{13}$$

Аксиома 3 (оценка сложности деревьев с идентичной структурой).

Если деревья $\hat{G}_1$ и $\hat{G}_2$ идентичны по структуре, $\hat{G}_1 \cong \hat{G}_2$, и суммарный вес дуг дерева $\hat{G}_1$ превышает суммарный вес дуг дерева $\hat{G}_2$, оценка сложности первого не может быть меньше оценки сложности второго

$$\sum_{k=1}^{\left|\hat{G}_1\right|_2} \gamma_{1,k} > \sum_{k=1}^{\left|\hat{G}_2\right|_2} \gamma_{2,k} \Rightarrow \theta\left(\hat{G}_1\right) \geq \theta\left(\hat{G}_2\right). \tag{14}$$

Аксиома 4 (о соотношении сложности дерева и поддерева).

Сложность любого поддерева $\hat{G}' \subset \hat{G}$ не превышает сложности дерева, в состав которого оно входит

$$\hat{G}' \subseteq \hat{G} \Rightarrow \theta\left(\hat{G}'\right) \leq \theta\left(\hat{G}\right). \tag{15}$$

В соответствии с принятыми аксиомами, для оценки сложности деревьев предлагается два варианта формул, причем каждый из вариантов согласован с набором аксиом (13)-(15):

Вариант № 1 – оценка сложности дерева по суммарному весу дуг

$$\theta^{(1)}\left(\hat{G}\right) = \sum_{k=1}^{\left|\hat{G}\right|_2} \gamma_k, \tag{16}$$

Вариант № 2 – оценка сложности дерева с учетом загруженности путей

$$\theta^{(2)}\left(\hat{G}\right) = \sum_{k=1}^{p} \sum_{j=1}^{|P_k|} \hat{\gamma}\left(P_{k,j}\right), \tag{17}$$

где p – общее число всех возможных путей из вершин, принадлежащих множеству экзогенных вершин $\underline{V}$ дерева $\hat{G}$ в вершины, принадлежащие множеству эндогенных вершин $\underline{V}$ (см. рисунок 5); множество путей в дереве $P = \left\{ P_k, k = \overline{1, p} \right\}, P_k = P_{k,1} \to P_{k,2} \to ... \to P_{k,|P_k|}$; $P_{k,j}$ – дуга, а $\widehat{\gamma}\left(P_{k,j}\right)$ – ее вес.

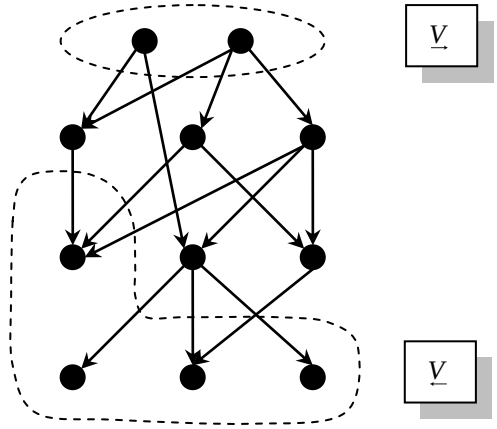


Рисунок 5. Иллюстрация к формуле (17)

Выбор критерия оценки сложности ОВС с древовидной структурой зависит от характера вычислений:

1) ОВС S представляет собой древовидный расчетный модуль, исполняемый *независимыми вычислительными устройствами*. В таких случаях пригодна оценка $\theta^{(1)}\left(\hat{G}\right)$ – она является верхней оценкой вычислительной сложности. С другой стороны эта оценка может использоваться и для случая распределенных вычислений с *зависимыми вычислительными устройствами*, но уже не как оценка сложности, а как оценка общей вычислительной емкости ОВС; данную оценку можно использовать при проектировании схем распараллеливания – чем меньше вычислительная емкость, тем меньше стоимость самих вычислений, возникает задача минимизации $\theta^{(1)}\left(\hat{G}\right)$ на множестве вариантов схем распараллеливания.

2) оценка $\theta^{(2)}\left(\hat{G}\right)$ может найти применение в тех случаях, когда ОВС представляет собой совокупность вычислительных устройств с древовидной структурой, причем основная вычислительная нагрузка приходится не на арифметические операции, а на передачу больших массивов информации; оценка сложности пропорциональна общей загруженности каналов потоками

информации в течении стабильного периода с неизменной структурой вычислений; чем большее среднее количество информации, приходящееся на канал, тем он более загруженный и тем больше вклад этого канала в общую оценку сложности всей древовидной структуры; для таких случаев подходит формула (17).

6. Оценка сложности иерархических ОВС

Иерархические ОВС отличаются от древовидных ОВС тем, что в качестве вершин фигурируют бикомпоненты. С позиций теории графов структура иерархических ОВС представляет собой дерево сильно связных подграфов (ССП).

Для оценки сложности деревьев ССП (сокращенно ДССП) необходимо преобразовать ДССП в дерево обобщенных вершин – орграф Герца, найти вес обобщенных дуг и применить к полученному результату формулы для оценки сложности деревьев вершин. Ранее была принята концепция, согласно которой сложность вычислений, сосредоточенных в дуге, ассоциируется с весом этой дуги, а вершины отождествляются с узлами распределения данных[7] (серверами данных). Распространяя допущение на ДССП, вполне логично ассоциировать сложность отдельной ССП с весом обобщенной дуги выходящей из ССП. В том случае, когда ССП имеет несколько исходящих из нее обобщенных дуг, вес каждой уменьшается во столько раз, сколько обобщенных дуг выходит из обобщенной вершины.

На концептуальном уровне для перехода от дерева ССП к дереву вершин необходим метод преобразования дерева ССП в орграф Герца, а для этого потребуются:

1) метод однозначной конкретизации ССП;
2) метод оценивания сложности ССП;
3) метод взвешивания обобщенных дуг.

В итоге будет сформировано дерево обобщенных вершин, сложность которого оценивается по правилам, сформулированным ранее в пункте 5.

6.1. Метод однозначной конкретизации ССП

Необходимо построить квазитреугольную форму матрицы смежности $\ddot{\mathbf{X}} : \ddot{\mathbf{X}} = con(G), \ddot{\mathbf{X}} :=: \underline{G}, G \approx \underline{G}, \ddot{\mathbf{X}} :=: \mathbf{X}$, где $\mathbf{X}$ – традиционная, не взвешенная

[7] Альтернативой является нагрузка на вершины в виде потенциала вершин. На наш взгляд, такой подход может сильно усложнить оценку сложности, особенно, когда орграф является сильно связным. Любые вычисления, связанные с оценками сложности, при таком подходе подразумевают решение системы уравнений Кирхгоффа.

матрица смежности исходного орграфа G. Для этого обратимся к методу построения процедурных моделей на основе операторных уравнений [1].

Процедурная модель построения квазитреугольной формы $\ddot{\ddot{\mathbf{X}}}$ матрицы смежности $\mathbf{X}$ верифицируется подстановкой в следующее операторное уравнение (если она записана в термах языка спецификаций, разумеется)

$$\begin{aligned}
&①: \lambda(\beta) \equiv \left\{ \left(bic\left(\tilde{G}_i \setminus \tilde{D}_i\right) = 1, i = 1..\beta \right) : \left(\bigcup_{i=1}^{\beta} \left(\tilde{G}_i \cup \tilde{D}_i\right) = G \right) \right\} = 1 \quad \wedge \\
&\forall \beta_1 < \beta : \lambda(\beta_1) = 0, \tilde{D}_i := \tilde{G}_i, \tilde{D}_i = \left(d_j^{(i)}, j = 1..\Upsilon_i \right), \ddot{\ddot{\mathbf{X}}}^T := \bigcup_{i=1}^{\beta} \left(\tilde{G}_i \cup \tilde{D}_i\right), \qquad (18) \\
&②: \tilde{G}_{i_1} \prec \tilde{G}_{i_2} \prec ... \prec \tilde{G}_{i_\beta} : \forall j \in \left\{ \overline{1,\beta\text{-}1} \right\} : \nexists l > j : \exists k, \left[\left(d_k = \left(v_{k_1} \to \circ \right) \right) : v_{k_1} \in \tilde{G}_{i_j} \right],
\end{aligned}$$

где $\lambda(\beta)$ – λ-предикат, по которому осуществляется проверка операторного уравнения на тождество. Использование λ-предиката в данном случае оправдано тем, что первое условие содержит один и тот же предикат дважды и он единственный в этом условии; $bic(G)$ – предикат, равный 1, если орграф G бикомпонентой (сильно связный орграф); «\» – операция вычитания множеств; Υ_i – дуговая размерность подмножеств дуг, образующих подматрицы в квазитреугольной форме $\ddot{\ddot{\mathbf{X}}}$; $\tilde{G}_{i_1} \prec \tilde{G}_{i_2} \prec ... \prec \tilde{G}_{i_\beta}$ – условие лексиграфического построения $\ddot{\ddot{\mathbf{X}}}$, а именно – над-квазидиагональ должна быть нулевой.

Если процедурная модель построения матрицы $\ddot{\ddot{\mathbf{X}}}$ и соответствующих этой матрице ССП $\tilde{G}_i$, совместно с кортежами исходящих дуг $\tilde{D}_i, i = 1..\beta$, записана в виде традиционного алгоритма, то его верификация по условию (18) заключается в следующем (см. иллюстрацию на рисунке 6):

- формируется достаточно представительная[8] выборка исходных орграфов $G : bic(G) = 0$;
- ко всем орграфам из этой выборки применяется процедура выявления ССП, и результат записывается в массив кортежей $\tilde{G}_i, i = 1..\circ$;
- каждый из элементов полученного массива проверяется на выполнение условий ① и ②;

[8] Вопросы оценки степени полноты выборки орграфов для верификации процедурных моделей весьма интересны.

- если окажется, что оба условия выполнены для каждого элемента массива, делается вывод об успешной экспериментальной верификации[9].

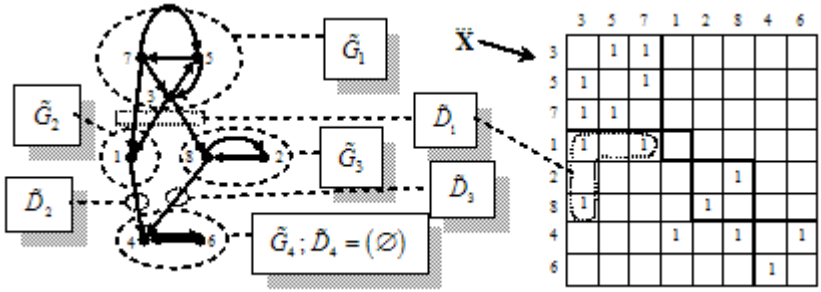


Рисунок 6. Иллюстрация к операторному уравнению (18)

Рассмотрим подробности (18) – сначала уравнение ①, состоящее из двух частей, объединенных логическим «И». В первом принимает участие λ-предикат и проверяется равенство $\lambda(\beta)=1$. Аргумент β предиката $\lambda(\beta)$ – число ССП в орграфе G. Каждой ССП соответствует кортеж исходящих дуг $\tilde{D}_i$, связывающих *i*-тый ССП с другими ССП ниже по иерархии; кортеж $\tilde{D}_i$ может быть пустым. Объединение пар кортежей $\tilde{G}_i \cup \tilde{D}_i$ должно быть равно исходному орграфу, в этом случае предикат $\lambda(\beta)$ = 1. Таким образом, предикат $\lambda(\beta)$ выражает требование целостности декомпозиции орграфа на ССП. Продолжим рассмотрение ①. Справа от логического «И» (знак «$\wedge$») находятся следующие условия, перечисленные через запятую, что, в соответствии со стандартами языков спецификаций, означает «перечисление в контексте логического "И"»: 1) $\forall\beta_1 < \beta : \lambda(\beta_1) = 0$ – требование максимальности β; 2) $\tilde{D}_i :=: \tilde{G}_i$ – каждый из кортежей исходящих дуг соответствует своему ССП, но не наоборот (нас интересуют дуги исходящие, а не входящие); 3) $\ddot{\mathbf{X}}^T :=: \bigcup_{i=1}^{\beta}\left(\tilde{G}_i \cup \tilde{D}_i\right)$ – квазитреугольная форма матрицы смежности представлена в транспонированном виде. Рассмотрим вторую часть условия (18). В ней содержится требование иерархичности ССП, а именно отсутствие дуг, исходящих по направлению, противоположному направлению иерархии, т.е. снизу-вверх. Была предложена процедурная

[9] Если считать допустимым термин «экспериментальная верификация». Фактически это экспериментальная апробация языка спецификаций на стадии его разработки.

модель, она была экспериментально верифицирована по (18). В ее основе – сортировка строк матрицы достижимости в антилексиграфическом порядке.

Матрица достижимости орграфа G определяется как $\mathbf{H}=\left(h_{ij}\right)_n; h_{ij}=\left(\exists k_l, l=1..\circ : \exists\left(i\to k_1\right),\left(i\to k_2\right),...,\left(i\to k_\circ\right)\right)\Rightarrow 1\backslash 0$, она используется для построения вектора $\underline{\mathbf{h}}=\left(\underline{h}_i\right)_{n\times 1}$, состоящего из строк бит $\underline{h}_i=\left(\begin{matrix}n\\ \bullet\\ j=1\end{matrix}\left(""+h_{ij}\right)\right).\left(""\overset{\varsigma(n)}{\oplus}\left(n-i+1\right)\right)$, где $\varsigma\left(n\right)=1+\left[\log_2 n\right]_0$ – число разрядов для хранения целого числа n в прямом двоичном коде; $\left[\circ\right]_0$ – целая часть числа; $\overset{\varsigma(n)}{\oplus}$ – разносортная двуместная операция сцепления строки бит (слева) и целого числа (справа). Перед сцеплением целое число преобразуется в прямой двоичный код – строку длиной $\varsigma\left(n\right)$ бит. Так, например, если в орграфе 7 вершин, то $\varsigma\left(7\right)=3$ и, если номер вершины, к примеру, равен 2, итогом операции станет (0-1)-строка «110». Результатом антилексиграфической сортировки строк символов вектора $\underline{\mathbf{h}}$ является перестановка Ï[1..*n*] одновременно и строк, и столбцов транспонированной матрицы смежности. В результате перестановки образуется квазитреугольная форма, характерная тем, что элементы, расположенные выше квазидиагонали, нулевые.

На рисунке 7 приведен пример построения дерева ССП.

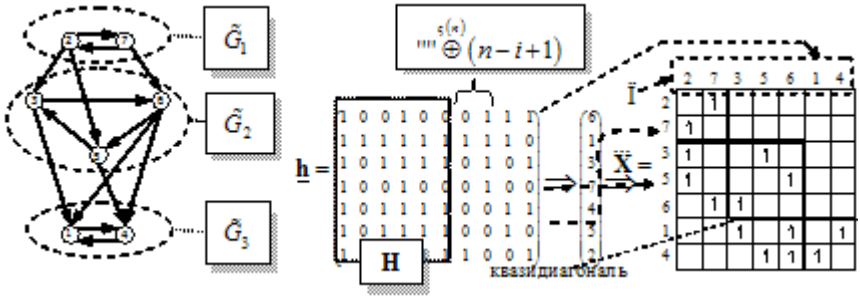


Рисунок 7. Иллюстрация к антилексиграфической сортировке вектора $\underline{\mathbf{h}}$

В этом примере орграф распадается на три ССП $G=bic\left(\tilde{G}_1\prec\tilde{G}_2\prec\tilde{G}_3\right)$, а соответствующее дерево имеет два функтора агрегирования [1]: $\chi_1\left(\gamma_{1,k_{1..3}}\right)\equiv\chi_1\left(\gamma\left(2\to 3\right),\gamma\left(2\to 5\right),\gamma\left(7\to 6\right)\right)$ и $\chi_2\left(\gamma_{2,k_{1..4}}\right)\equiv\chi_2\left(\gamma\left(3\to 1\right),\gamma\left(5\to 4\right),\gamma\left(6\to 1\right),\gamma\left(6\to 4\right)\right)$.Следует отметить, что

ССП, получаемые в результате антилексиграфической сортировки строк бит – компонент вектора $\underline{\mathbf{h}}$ – образуют иерархическую структуру.

Процедурная модель поиска ССП в новых обозначениях представляет собой отображение $Dec: G \to \left(\tilde{G}_i, \tilde{D}_i, i = 1..\beta\right)$. Предлагается следующая процедурная модель для поиска ССП

$$Dec = \left(\begin{array}{l} \ddot{I}: 0 + \underline{\mathbf{h}}_{\ddot{I}_j} \geq 0 + \underline{\mathbf{h}}_{\ddot{I}_{j+1}},\ j = 1..n-1 \Rightarrow \lambda(k) = 0 + \overset{n}{\underset{j=1}{\bullet}} h_{kj}, \\ \tilde{G}_i = \left(\begin{array}{l} \left(\begin{array}{l} \forall k \in \ddot{I}_{l..|\tilde{G}_i|}: \lambda(k) = idem; \\ l = 1 + \left[\left(0 :=: (i = 1)\right) \overline{\vee} \sum\limits_{j=1}^{i-1} \left| \tilde{G}_j \right| \right] \end{array} \right) \wedge \\ W_i = \overset{\circ}{\bigcup\limits_{k=1}} \left[\begin{array}{l} G\left(\left(v(\underline{d}_k), v\left(\overline{d}_k\right) \right), (d_k), \left(\gamma(d_k)\right) \right): \\ \underline{d}_k \in \tilde{G}_i, \overline{d}_k \notin \tilde{G}_i \end{array} \right] \end{array} \right), i = 1..\beta \end{array} \right). \qquad (19)$$

Рассмотрим выражение (19). Результатом вычислений в процедурной модели Dec является перестановка $\ddot{I}[1..n] = \left(\ddot{I}_1, \ddot{I}_2, ..., \ddot{I}_n\right)$ отрезка чисел натурального ряда от 1 до *n*. Если, следуя этой перестановке, одновременно переставить и строки, и столбцы транспонированной матрицы смежности, получим квазитреугольную форму $\ddot{\ddot{\mathbf{X}}}$, на основе которой обнаруживаются ССП $\tilde{G}_i$, наряду с набором кортежей исходящих дуг $\tilde{D}_i, i = 1..\beta$. В выражении (19) используется обобщенное «ИЛИ» – двуместная, разносортная операция $\overline{\vee}$ [1] и конкатенация строк в операции «.». Условие сравнения элементов, сортируемых в антилексиграфическом порядке, отражает неравенство $0 + \underline{\mathbf{h}}_{\ddot{I}_j} \geq 0 + \underline{\mathbf{h}}_{\ddot{I}_{j+1}}, j = 1..n-1$. Окончание строк $\left(\text{""} + \overset{\varsigma(n)}{\underset{i=1}{\bigoplus}} (n - i + 1) \right)$ в векторе $\mathbf{h}$ служит для лексиграфической нумерации вершин в рамках ССП, в соответствии с нумерацией вершин в изначальном орграфе. Напомним, что выражение 0+«строка бит» равно целому числу, полученному из прямого двоичного кода, записанного в строку бит слева направо, начиная со старшего бита. На рисунке 8 показан результат сортировки, он соответствует орграфу на рисунке 7. Показано как окончание строк бит в векторе $\underline{\mathbf{h}}$ помогает выстроить номера вершин в рамках ССП по возрастанию индексов: так, например, $\tilde{G}_1$ характеризуется вершинами $\left(v_2, v_7\right)$, а не $\left(v_7, v_2\right)$.

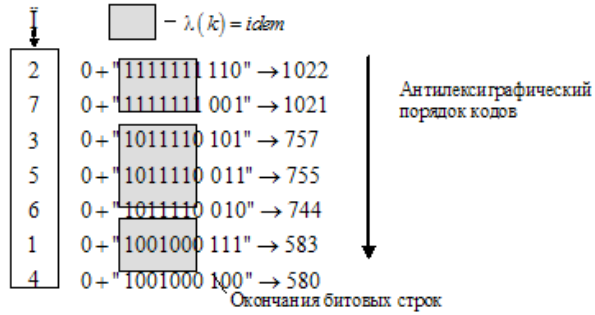


Рисунок 8. Иллюстрация к процедурной модели (19)

Заметим, что для нахождения ССП множество контуров орграфа не используется, знать необходимо только матрицу достижимости.

6.2. Метод поиска ССП при наличии точек сочленения в орграфе ОВС

В ряде случаев оценка вычислительной нагрузки с использованием оценок сложности затрудняется лепестковой топологией ОВС (рисунок 9), когда центральное место в системе отводится облачному серверу с защищенными данными, или когда в отдельную подсистему выделяется облачный сервер, непосредственно связанный одновременно с несколькими вычислительными подстанциями, имеющими сильно связную структуру, и, возможно, кластерами.

Рассмотрим орграф ОВС – он показан на рисунке 10 без разрываемой дуги, соответствующей глобальному итерационному циклу. Проблема структурирования, при котором облачный сервер, наряду с вычислительными подстанциями, объединяется в единую бикомпоненту, заключаются в том, что все три элемента рассматриваются как единое целое; итерационные циклы могут содержать в себе нежелательное ассоциирование облачного сервера с вершинами, относящимися к вычислительным подстанциям, как показано на рисунке 11. Вариант структуризации б) отличается от варианта а) тем, что взамен одной бикомпоненты $\tilde{G}_1$ обнаружены две контурные подсистемы $\dot{G}_1$ и $\dot{G}_2$. Контурные ССП (КССП) характеризуются тем, что каждая дуга КССП участвует в каждом контуре подмножества контуров, локализованного в конкретном КССП, которому принадлежит дуга, в то же время ни одна КССП не имеет пересечения по дугам с другим КССП. В частном случае бикомпонента может являться КССП, если в орграфе нет точек сочленения. Работой вычислительных подстанций, соответствующих этим ССП, управляют теперь уже два итерационных цикла $Ц_1$ и $Ц_2$, а точка сочленения ТС при этом не теряет своей значимости как координатор вычислений. В

данном случае оценка сложности $\theta\left(\dot{G}_1 \cup \dot{G}_2\right) = \theta\left(\dot{G}_1\right) + \theta\left(\dot{G}_2\right)$ – сумма оценок контурных ССП.

В ряде случаев облачный сервер может рассматриваться, как самостоятельная подсистема, тогда количество точек сочленения увеличивается. В случае, если облачный сервер подключен, как показано на рисунке 12, к двум подстанциям, естественно предположить, что точек сочленения именно две, т.е. их число равно количеству подстанций, сопряженных с сервером. Именно таким образом учитываются особенности ОВС при наличии крупных серверов раздачи вычислительной нагрузки. Ситуации, когда при структуризации ОВС в результатном орграфе возникают точки сочленения, возможны при организации грид-систем в рамках крупных сетевых ИС с несколькими кластерами, удаленными друг от друга территориально, когда магистральные линии нагружены весьма сильно, являясь линиями компьютерной связи общегеографического назначения.

Применительно к проблематике организации высокопроизводительных облачных вычислений, подобные ситуации возможны, например, при решении систем уравнений большой размерности, разделенных на два (и более) блока с одним (и более) уравнением на грани разделения (уравнение сопряжения). В таких случаях можно найти перестановку строк и столбцов матрицы смежности, применив которую эта матрица приобретет псевдо-квазидиагональный вид, как показано на рисунке 13.

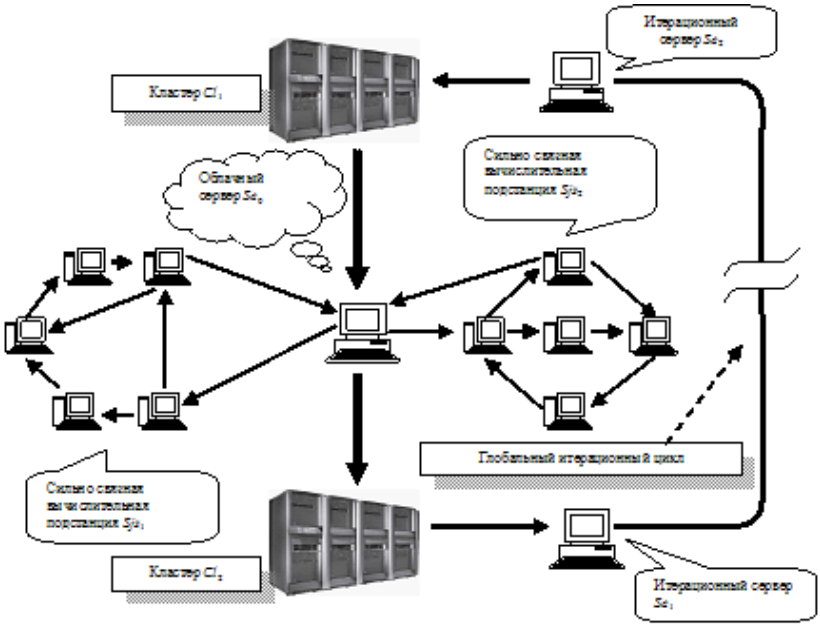


Рисунок 9. Пример ОВС с глобальными итерациями.

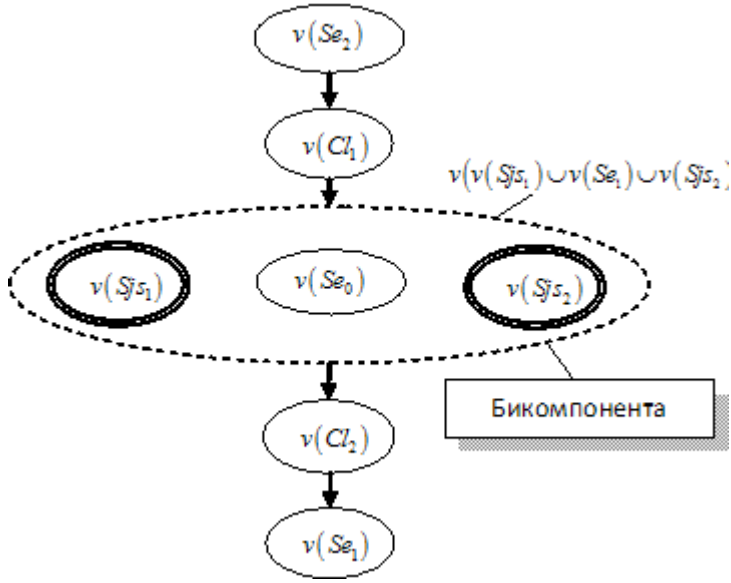


Рисунок 10. Структурирование грид-системы: случай одной бикомпоненты.

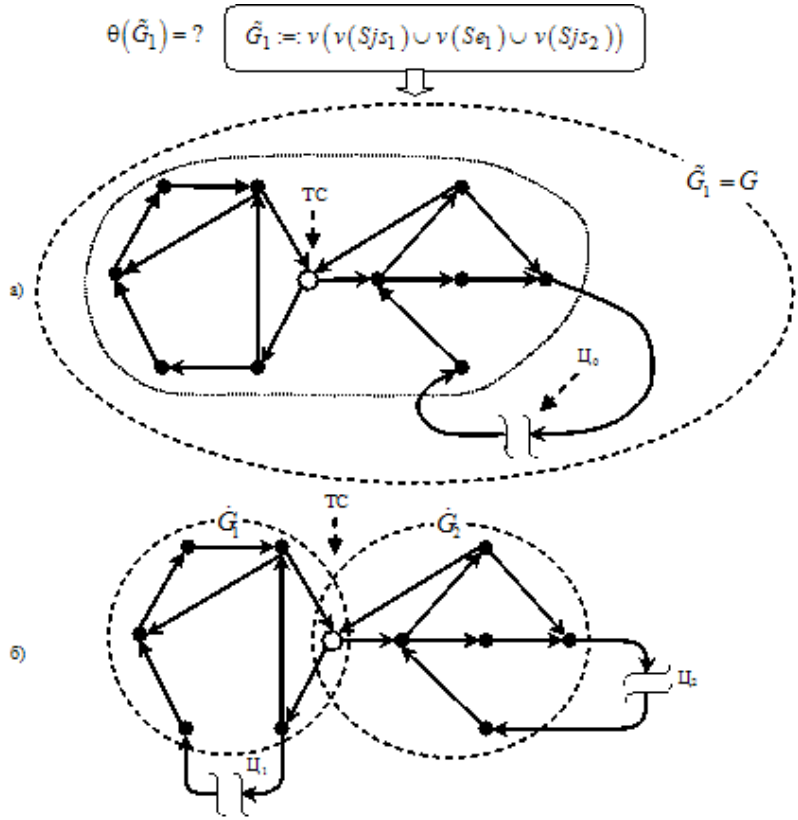


Рисунок 11. Варианты структуризации ОВС на рисунке 10: а) точка сочленения не учитывается; б) учитывается.

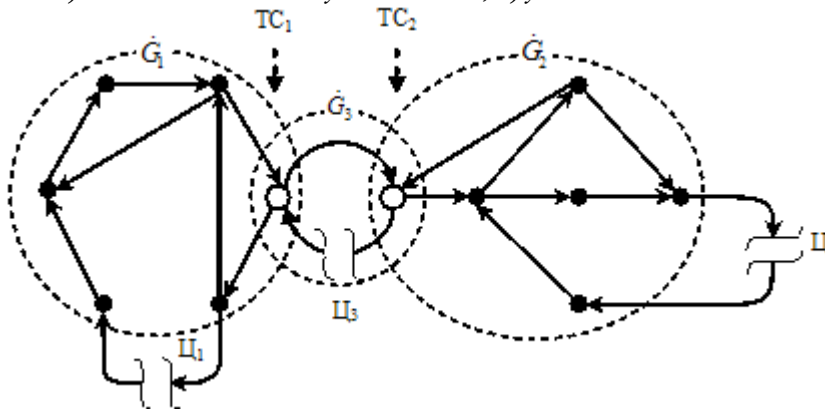


Рисунок 12. Вариант лепестковой структуризации с центральным облачным сервером.

Вообще говоря, бикомпоненты и контурные подграфы можно считать разнотипными сильно связными подграфами, и, по формальным соображениям, в орграфе могут одновременно существовать и те, и другие. Тем не менее, дифференцированный подход к перечислению подграфов (с разделением их на типы) значительно усложняет дальнейшую формализацию. Разумно приравнять бикомпоненты, в которых нет точек сочленения, к КССП, если используется букетная модель декомпозиции – в данном случае ОВС [1]. Выбор же конкретной процедурной модели декомпозиции орграфа на ССП зависит от конкретной ситуации, в которой сложность оценивается путем постепенного упрощения исследуемой системы с дальнейшей композицией оценок примитивных структур, восходя с нижнего уровня декомпозиции.

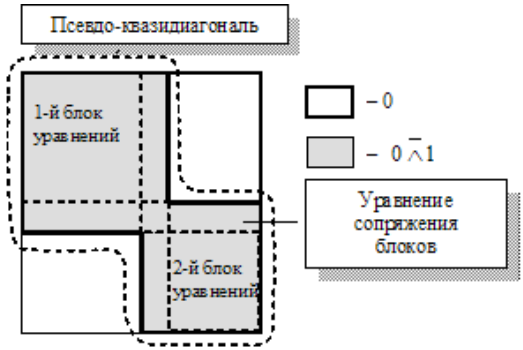


Рисунок 13. Матрица смежности для системы уравнений большой размерности с уравнением сопряжения блоков

Применительно к облачным вычислениям, в пользу метода контурных подграфов можно отнести изначальную структуру соединения облачных серверов. Так, например, если в схеме присутствуют соединения типа «звезда» и ставится задача оценки стабильности работы облачной системы, контурные подграфы более предпочтительны, чем бикомпоненты. Рассмотрим процедурную модель поиска контурных ССП, основанную на принципах «букетов». За основу представления контуров удобнее взять последовательность дуг, а не вершин, в этом случае кортеж дуг – контур – лучше согласуется с контурными ССП. Пусть $C(G) = \left(C_i(G) := \left(d^{(i)}_{1..\eta_i} \middle| \lambda\left(d^{(i)}_{1..\eta_i}\right) = 1 \right), i = \overline{1,K} \right)$ – кортеж контуров орграфа G, где K – число контуров; в i-том контуре содержится η_i дуг, $\lambda\left(d^{(i)}_{1..\eta_i}\right)$ – λ-предикат, в который вложено определение «элементарный контур», а именно

$$\lambda\left(d^{(i)}_{1..\eta_i}\right)=\left(\begin{array}{l}\left(\overline{d}^{(i)}_k=\underline{d}^{(i)}_{k+1},k=\overline{1,\eta_i-1}\wedge\underline{d}^{(i)}_1=\overline{d}^{(i)}_{\eta_i}\right)\wedge\\ \forall v_\circ\Bbbk\ \ \overline{d}^{(i)}_k,k\in\left\{\overline{1,\eta_i-1}\right\}:\nexists j\neq k+1:v_\circ=\underline{d}^{(i)}_j\wedge\\ \underline{d}^{(i)}_1<\underline{d}^{(i)}_{2..\eta_i}\end{array}\right)=>1\subsetneq 0,\qquad(20)$$

где $\underline{d}^{(i)}_k$ – порядковый номер вершины в соответствующем кортеже (вершин), из которой исходит *k*-тая дуга, а $\overline{d}^{(i)}_k$ – наоборот, порядковый номер вершины, в которую входит эта дуга. Процедурная модель поиска контурных подграфов является решением системы из трёх операторных уравнений

$$\text{①}:\lambda\left(\dot{\beta}\right)\equiv\left\{\left(bic\left(\dot{G}_i\right)=1,i=1..\dot{\beta}\right):\left(\bigcup_{i=1}^{\dot{\beta}}\dot{G}_i=G\right)\right\}=1\ \wedge\forall\dot{\beta}_1<\dot{\beta}:\lambda\left(\dot{\beta}_1\right)=0,$$

$$\text{②}:\forall j,l\in\left\{\overline{1,\dot{\beta}}\right\},l\neq j:C_{1..\eta_j}\left(\dot{G}_j\right)\cap C_{1..\eta_l}\left(\dot{G}_l\right)=\varnothing,\qquad(21)$$

$$\text{③}:\dot{G}_{i_1}\prec\dot{G}_{i_2..i_{\dot{\beta}}}:\forall j\in\left\{\overline{1,\dot{\beta}\text{-}1}\right\},\forall l\in\left\{\overline{2,\dot{\beta}}\right\},l>j:\overline{C_i\left(\dot{G}_l\right)}>\overline{C_i\left(\dot{G}_j\right)}.$$

В уравнении «①» содержатся: требование сильной связности подграфов, условие полноты (объединение подграфов равно исходному орграфу G) и рекурсивное условие $\forall\dot{\beta}_1<\dot{\beta}:\lambda\left(\dot{\beta}_1\right)=0$, по которому число искомых подсистем было максимальным. В уравнении «②» записано дополнительное требование к контурным ССП: они должны быть не только сильно связными (следствие наличия в них контуров, в которых участвуют все вершины ССП), но и не должны иметь общих дуг. Наконец, в «③» представлено правило предшествования контурных ССП, согласно которому устанавливается их взаиморасположение. Для контурных ССП оно заключается в следующем: подграф $\dot{G}_j$ встречается в списке левее подграфа $\dot{G}_l$, если $\overline{C_i\left(\dot{G}_j\right)}$ – максимальный по значению индекс вершины *j*-го подграфа – не превышает аналогичного индекса в подграфе с номером *l*.

7. Метод структурно-параметрической минимизации орграфа ОВС

Орграф ОВС может содержать элементы, способствующие увеличению времени вычисления оценки сложности, что негативно сказывается на применении этих оценок в режиме online. К таким элементам, мешающим анализу сложности, относятся ветви вычислений, выполняемых последовательно, и скрытые параллельные ветви вычислений. На рисунке 14

показаны: пример орграфа (*G*), содержащего избыточные элементы, и результат структурно-параметрической минимизации (СПМ) – орграф *G'*.

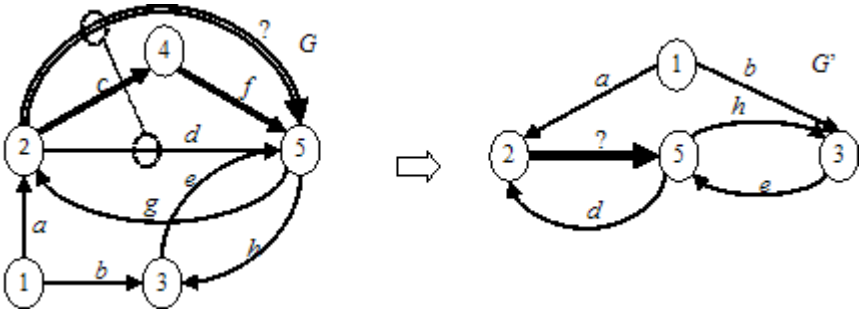


Рисунок 14. Пример орграфа ОВС с избыточными элементами

Транзитивный вычислительный поток 2→4→5 (дуги выделены утолщенными линиями) заменяется обобщенной дугой $2 \Rightarrow 5$, ее параметричность показана на рисунке 14 в виде вопросительного знака – требуется определить, каким образом она будет оцениваться. Через обобщенную дугу (такие дуги выделены двойной линией) $2 \Rightarrow 5$ проходит вычислительный поток, выполняемый параллельно потоку вычислений в изначальной дуге 2→5 орграфа ОВС (отмечено кружочками и соединяющей их точечной линией): у этих дуг совпадают начальная и конечная вершины.

Оценка сложности параллельных вычислительных потоков должна учитывать характер их выполнения: если эти потоки обрабатываются в параллельном режиме, оценка сложности должна быть согласована с критической линией, сложность которой максимальна. Есть еще одно важное обстоятельство – при замене транзитивной ветви обобщенной дугой в орграфе *G'* не должна появиться реверсивная петля. Обобщенную дугу d^*, возникающую в результате стягивания транзитивной ветви, определяем как решение операторного уравнения

$$d^* := \left(\left(\underline{d}^* \to \overline{d}^* \right) = con \left(\begin{array}{l} \underset{i=1}{\overset{m^*}{\rightarrow}} \left(d_i \in D^* : E\left(D^*\right) = 1 \right), \\ \gamma^* = \min\limits_{1 \le i \le m^*} \gamma\left(d_i\right), \underline{d}^* = d_1 \in D^*, \overline{d}^* = d_{m^*} \in D^* \end{array} \right) \right), \qquad (22)$$

где утверждается, что обобщенная дуга представлена начальной вершиной $\underline{d}^*$ и конечной вершиной $\overline{d}^*$, дуга конкретизируется последовательностью сцепленных дуг, параметричность обобщенной дуги минимальна, а начало и конец совпадают с начальной вершиной первой и конечной вершиной последней дуги в кортеже D^*; функтор *E* определяет минимальный по

мощности кортеж дуг $\breve{D}^*$, без которых кортеж D^* становится транзитивной ветвью

$$E\left(D^*\right)=\left(\exists\breve{D}^*\in D^*:E\left(D^*\setminus\breve{D}^*\right)=1\wedge\left|\breve{D}^*\right|=\min_{\circ}\left|\breve{D}^*\right|\right)\Rightarrow\breve{D}^*\subsetneq\varnothing. \quad (23)$$

Для процедурной модели СПМ потребуется предикат, который выявляет скрытую параллельность обобщенной дуги d^* орграфа G и изначальной дуги $d_k\in D, k=1..m$

$$Q\left(d^*,d_k\right)=\left(\underline{d}_1^*=\underline{d}_k\wedge\overline{d}_{m^*}^*=\overline{d}_k\right)\Rightarrow 1\subsetneq 0. \quad (24)$$

Склеивание параллельных дуг оформим в виде двуместной операции «$\otimes$», а именно

$$d^*\otimes d_k:\left(d^*,d_k\right)\to\left(\left(\underline{d}_k\to\overline{d}_k\right)=con\left(\left(d^*,d_k\right)\overline{\wedge}\left(\gamma^*:=\gamma^*+\left(\gamma_k=des\left(d_k\right)\in G'\right)\right)\right)\right) \quad (25)$$

Следует отметить, что 1) перед СПМ матрица контуров $\breve{\mathbf{C}}\left(G\right)=\left(\breve{c}_{ij}\left(G\right)\right)_{K\times m}$ должна быть известна; 2) столбцы матрицы, соответствующие транзитивной ветви, одинаковы, т.к. все ее дуги входят в одни и те же контуры орграфа. Это свойство позволяет ускорить поиск транзитивных дуг. Однако условие равенства столбцов в данном случае не является достаточным, что как раз и учитывает функтор $E\left(D^*\right)$: его работа иллюстрируется на рисунке 15.

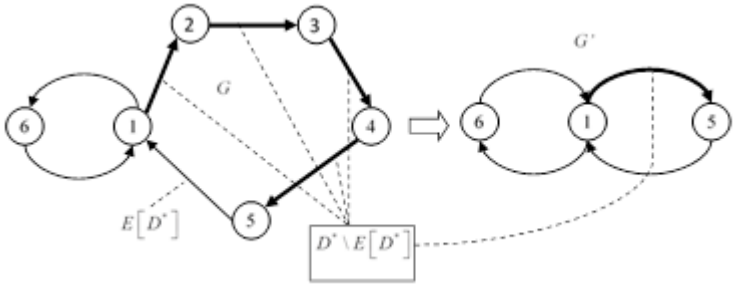


Рисунок 15. Иллюстрация к функтору $E\left(D^*\right)$.

Сформулируем процедурную модель метода СПМ как рекурсивный алгоритм.

1°. Вход StrMin(*G*): $G=\left(V,D,\Gamma\right)$ – орграф ОВС; $\breve{\mathbf{C}}\left(G\right)=\left(\breve{c}_{ij}\left(G\right)\right)_{K\times m}$ – матрица контуров, соответствующая орграфу *G*.

2°. Инициализируем возвращаемое значение StrMin – орграф $G' \coloneqq G$.

3°. Обнуляем кортеж дуг, составляющих транзитивную ветвь: $D^* \coloneqq \varnothing$, $m^* \coloneqq \varnothing$.

4°. Инициализируем булев массив $pD^* = \left[pD_j^* \coloneqq 0, j = 1..m \right]$. Если $pD_j^* = 1$, столбец с номером *j* считается «рассмотренным» и пропускается.

5°. Цикл по не «рассмотренным» столбцам матрицы $\breve{\mathbf{C}}(G)$, $j = (1..m) \wedge pD_j^* = 0$.

6°. Цикл по остальным столбцам матрицы $\breve{\mathbf{C}}(G)$, кроме столбца с номером j: $\nu = 1..j-1..j+1..m \wedge pD_\nu^* = 0$.

7°. Проверяем идентичность столбов $\breve{c}_{ij} = \breve{c}_{i\nu}, i = 1..K$? Если они идентичны, добавляем дугу в кортеж $D^* \coloneqq D^* \cup \left\{ d^*_{++m^*} \coloneqq d_{D^*=\varnothing \Rightarrow j \subsetneq \nu} \right\}$ и помечаем соответствующий столбец как «рассмотренный»: $pD^*_{D^*=\varnothing \Rightarrow j \subsetneq \nu} = 1$.

8°. Конец цикла по ν.

9°. Конец цикла по j.

10°. Тестируем найденный кортеж на транзитивность $E(D^*) \neq \varnothing$? Если тестирование прошло удачно, переходим на п. 12°.

11°. Применяем функтор $E(D^*)$: если $E(D^*) \neq \varnothing$, полагаем $D^* \coloneqq D^* \setminus E(D^*)$, в противном случае процедурная модель заканчивает свою работу с кодом удачи.

12°. Исключаем транзитивную ветвь из результатного орграфа $G' \coloneqq G' \setminus \lambda G(\circ, D^*, \circ)$ (здесь $\lambda G(\circ, D^*, \circ)$ – временный орграф, в котором множество дуг представлено множеством D^*) и находим обобщенную дугу

d^* из решения операторного уравнения (22), добавляем информацию в результатный орграф $G' := G' \cup \lambda G\left(\circ, \left\{d^*\right\}, \circ\right)$.

13°. Проверяем наличие скрытой параллельной ветви, открываем цикл по перебору дуг орграфа G', всех, кроме d^*, $k = 1..\left|G'\right|_2, d_k \neq d^*$.

14°. Дуга d_k содержит параллельные вычисления с обобщенной дугой d^*? т.е. $Q\left(d^*, d_k\right) = 1$? Если «да», то склеиваем дуги $d^* := d^* \otimes d_k$.

15°. Конец цикла по k.

16°. Проверяем $G' \neq G$? Если «да», то необходимо продолжить рекурсию: $G' := \mathrm{StrMin}\left(G'\right)$.

17°. Выход – орграф G'.

8. Оценка сложности сильно связных вычислительных примитивов

Ранее, в п. 2, было сделано допущение о трансформации оценок вычислительной нагрузки ОВС в оценки сложности объектов класса «орграф». Как следствие, в данном разделе термин «сильно связный вычислительный примитив» будет заменен «простейшим сильно связным орграфом» в том смысле «простейшим», что формулы оценки сложности выводятся без использования каких-либо специальных алгоритмов и программ.

Оценка сложности орграфа подразумевает постепенное упрощение, вплоть до столь примитивных структур, для которых оценка $\theta\left(G\right)$ выведена заранее в виде формулы. Таким образом, в процессе упрощения, вместо повторного анализа ранее проанализированных структур, используются накопленные знания с готовыми ответами. Первыми в списке исследованных и готовых к занесению в базу знаний должны быть сильно связные орграфы небольшой размерности – из 2-х и 3-х вершин. Для них потребуется вывести формулы оценки сложности.

Инвариант $R\left(G\right)$ позволяет упорядочить список сильно связных орграфов, являющихся отображением наиболее простых топологий итерационных и циклических вычислений. Из этого списка надо удалить изоморфные орграфы, а для оставшихся орграфов найти формулы оценок сложности $\theta\left(G\right)$.

Следует заметить, что в инварианте $R(G)$ вес дуг не используется и $\forall G: G = \underline{G} \Rightarrow G \coloneqq G \setminus \Gamma$. Вне вопросов индексации будем использовать восстановленное $G \coloneqq G \cup \Gamma$ (при этом суть рассуждений не меняется и не теряется, хотя для практической реализации этот аспект весьма важен).

Рассмотрим коды всех возможных сильно связных орграфов с тремя вершинами, ранжируя их в порядке возрастания инварианта $R_0(G)$: 23, 25, 27, 29, 31, 38, 39, 45, 46, 47, 54, 55, 57, 58, 59, 61, 62, 63. В силу своей уникальности, критерий $R_0(G)$ взят за основу построения диаграмм изоморфизмов для класса сильно связных орграфов заданной вершинной размерности $|G|_1$. Стрелки на этих диаграммах идут справа-налево, упираясь в коды базовых орграфов. На рисунке 16 показаны результаты исследования изоморфизма вышеупомянутого списка орграфов по критерию $R_0(G)$, в виде диаграммы изоморфизмов. «Сбой» по дуговой размерности $|G|_2$ наблюдается сразу же, начиная с кода 23 – ему соответствует орграф с 4-мя дугами, тогда как коду 25 соответствует орграф с тремя дугами: налицо нарушение принципа назначения инвариантов. Серым фоном на рисунке 15 отмечены базовые коды, т.е. соответствующие изоморфные орграфы имеют код $R_0(G)$, значение которого превышает базовый код.

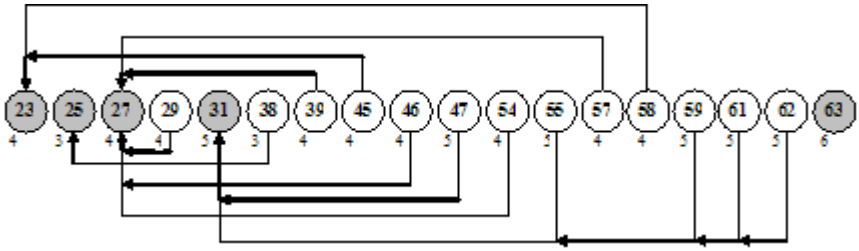


Рисунок 16. Коды $R_0(G)$ орграфов с тремя вершинами и число дуг (внизу)

9. Сложность диполя

Диполь – сильно связный орграф, примем для него обозначение в виде $\overset{(3)}{G}$, где верхний индекс “3” – значение инварианта $R_0\left(\overset{(3)}{G}\right)$. В диполе две вершины и две дуги:

$$\overset{(3)}{G} = (V, D, \Gamma), V = (v_1, v_2), D = (d_1, d_2),$$
$$d_1 = (v_1 \to v_2), d_2 = (v_2 \to v_1), \Gamma = (\gamma_1 :=: d_1, \gamma_2 :=: d_2). \quad (26)$$

Диполь $\overset{(3)}{G}$ показан на рисунке 17. Он является математическим описанием структуры простейшей ОВС. Вершины диполя соответствуют блокам распределения информации (серверам), а дуги – вычислительным ресурсам (компьютерам, кластерам, облачным площадкам).

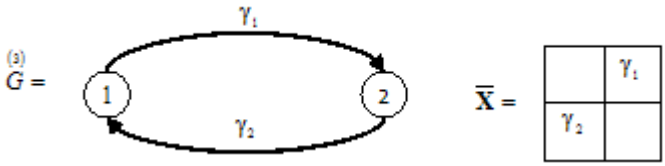


Рисунок 17. Диполь и его взвешенная матрица смежности

Оценка сложности систем производится путем серии упрощений, осуществляемых однотипно. В когнитивном подтексте при вычислении оценки

На рисунке 17 легко заметить и вполне очевидно сложности серию упрощений можно представить в виде дерева рекурсии.

Структура исходного состояния ОВС представлена сильно связным орграфом $G \neq \overset{(3)}{G}, bic(G) = 1$. Требуется оценить вычислительную сложность ОВС. Для этого исходный орграф подвергается поэтапной декомпозиции. Начальное состояние орграфа считается нулевым этапом. Применительно к ОВС именно диполь является конечной структурой в дереве рекурсивного вычисления оценки сложности.

Найдя способ перевода диполя в разомкнутое состояние, и, одновременно, производя при этом оценку сложности, можно оценить сложность всей ОВС в целом.

наличие двух альтернативных вариантов размыкания диполя: либо разрыв дуги (1→2), либо разрыв дуги (2→1). В ОВС разрыв любой дуги сопряжен с необходимостью итерирования по одной или нескольким переменным, передаваемым в результате распределения информации в вершинах для последующего расчета в дугах орграфа. Для разрешения альтернативы надо принять во внимание, что дуги являются топологическим отображением вычислительного процесса, и выбор, по какой именно дуге будет осуществляться итерирование, определяется сложностью дуг и их сочленением в орграфе ОВС. При этом надо сразу оговориться, что перед расчетами общей оценки сложности всей ОВС в целом параметричность всех без исключения дуг орграфа должна быть заранее известна.

Общая вычислительная сложность диполя – она должна быть минимальна, на этом строится общая оценка сложности орграфа ОВС.

На основе анализа экспериментальной информации, с привлечением ряда аксиом сложности в итоге была нами предложена следующая формула

$$\theta\left(\overset{(3)}{G}\right) = \min\left\{\gamma_1\left(1+\gamma_2\right), \gamma_2\left(1+\gamma_1\right)\right\}. \qquad (27)$$

10. Процедурная модель оценки вычислительной сложности ОВС

Для оценки вычислительной сложности ОВС разработана рекурсивная процедурная модель, в основе которой – принципы построения мультипликативных оценок сложности, когда основой построения шкалы сложности орграфов является вложенность вычислений в итерационные циклы. В описании модели используется термин «контурион» – это список дуг орграфа, ранжированных в порядке убывания их контурности, а в каждом подмножестве дуг, характерных одинаковой контурностью, дуги ранжируются по возрастанию параметричности [1]. В таком случае дуги, имеющие наибольший приоритет к разрыву, стоят в начале списка. Для уменьшения числа вариантов разрыва применяется достаточно простой прием, относящийся к методике «ветвей и границ»: перебор прекращается, если минимум сложности, найденный для очередного подмножества, характеризуемого одинаковой контурностью, по отношению к предыдущему подмножеству стал возрастать.

1. Вход (рекурсивный адаптер): орграф $G = (V, D, \Gamma)$.
2. Протокольная служба[10] активирована? Нет – посылка сообщения на активизацию.
3. Орграф сильно связный? $bic(G) = 1$? «Да» – на п. 12.
4. Структурная декомпозиция: разбиваем G на $\text{ССП}_i, i = 1..\beta$.
5. $\theta_0 = 0$.
6. Цикл по ССП, $i = 1..\beta$.
7. Вход (рекурсивный адаптер): орграф $G = \text{ССП}_i$.
8. $\theta_0 = \theta_0 + \theta$.
9. Конец цикла по i.
10. $\theta = \theta_0$.
11. Выход: θ.

[10] Отдельная подзадача, работающая в фоновом режиме; формирует протокол разрывов и осуществляет взаимодействие с базой знаний оценок сложности.

12. С точностью до изоморфизма орграф G имеется в базе знаний оценок сложности (база знаний индексирована по инвариантам сильно связных орграфов)? «Да» – обращаемся с запросом в базу знаний и получаем оценку $\theta \equiv \theta(G)$, выходим из рекурсивного адаптера со значением θ.
13. Проводим СПМ орграфа $G = \mathrm{StrMin}(G)$.
14. Построение контуриона на основе $\breve{\mathbf{C}}(G)$ и Γ.
15. $\theta_0 = HUGE\_VAL$ (наибольшее из всех возможных вещественных чисел в языке C++).
16. Цикл по дугам контуриона, $i = 1..m$.
17. Разрыв дуги $G' = G \setminus \{d_i\}$.
18. Декомпозиция G' на контурные подсистемы.
19. Цикл по списку контурных подсистем с суммированием результатов оценки сложности. Полагаем, что очередная КССП имеет обобщенный алиас G'.
20. Вход (рекурсивный адаптер): G = орграф G'
21. Уточняем мультипликативную оценку сложности $\theta = \gamma_i(1 + \theta)$.
22. $\theta < \theta_0$? Оценка убывает? «НЕТ» - выходим из цикла по *i*.
23. Запоминаем оценку $\theta_0 = \theta$.
24. Конец цикла, открытого в п. 19.
25. Конец цикла по *i*.
26. Выход со значением $\theta = \theta_0$.

11. Заключение

Разработан теоретический базис оценки сложности крупноблочных облачных вычислений, использующих арифметику повышенной точности, включающий в себя методы и алгоритмы, предназначенные для проектирования ОВС. В настоящее время работа ведется в направлении дальнейшей конкретизации оценок сложности реальных вычислительных диполей, проводятся вычислительные эксперименты по использованию арифметики повышенной точности в типовых численных методах с дальнейшей аппроксимацией полученных результатов тестирования. Планируется получить конкретные выражения для функционалов параметричности дуг орграфа ОВС при решении крупноблочных задач математического моделирования.

Список литературы

[1]. Подольский, В.Е. Повышение эффективности региональных образовательных компьютерных сетей с использованием элементов структурного анализа и теории сложности / В.Е. Подольский, С.С. Толстых. – М.: Машиностроение, 2006. – 176 с.

[2]. Подольский, В.Е. Использование критериев структурной сложности для имитационного моделирования региональных компьютерных сетей / В.Е. Подольский, С.С. Толстых // Параллельные вычисления в задачах математической физики: Сб. ст. – Ростов н/Д: Изд-во РГУ, 2005. – С. 67 – 75.

[3]. Подольский, В.Е. Оптимизация кластерных вычислений с использованием критериев структурной сложности / В.Е. Подольский, С.С. Толстых // Вторая Сибирская школа-семинар по параллельным вычислениям. – Томск: Изд-во Том. ун-та, 2004. – С. 45-50.

Evaluation of complexity of the large-block cloud computing using arithmetic with enhanced accuracy

S.S. Tolstyh <inf@tstu.ru>
V.E.Podolsky <director@director.tixmcnit.tambov.su>
Tambov State Technical University, Tambov, Russia

Abstract. In article questions of an evaluation of complexity of large-block cloud computing with enhanced accuracy are considered. This development is directed on the decision in a cloud of tasks of mathematical simulation with special requirements of accuracy. In particular it is about obtaining the precision and confidential solution of tasks with complex couplings between subtasks in the form of large blocks and the score time considerably exceeding transmission time of information in between. The methodology of an evaluation of complexity of tasks of this sort used for creation of the computing systems, optimum on productivity functioning in the cloudy environment is offered.

Keywords. large-block parallel computing, cloud computing, precision computation, precision and confidential solution of tasks, floating point arithmetic of enhanced accuracy.

References

[1]. Podolskiy V.E., Tolstyh S.S. Povyishenie effektivnosti regionalnyih obrazovatelnyih kompyuternyih setey s ispolzovaniem elementov strukturnogo analiza i teorii slozhnosti [Increase of efficiency of regional educational computer networks with use of elements of the structural analysis and theory of complexity]. Moscow, Mashinostroenie, 2006. 176 p (in Russian).

[2]. Podolskiy, V.E. Ispolzovanie kriteriev strukturnoy slozhnosti dlya imitatsionnogo modelirovaniya regionalnyih kompyuternyih setey / V.E. Podolskiy, Tolstyh S.S. [The use of criteria of the structural complexity for simulation of regional computer networks] // Parallelnyie vyichisleniya v zadachah matematicheskoy fiziki: Sb. st. – Rostov n/D: Izd-vo RGU, 2005. – S. 67 – 75. (In Russian)

[3]. Podolskiy, V.E. Optimizatsiya klasternyih vyichisleniy s ispolzovaniem kriteriev strukturnoy slozhnosti [Optimization of cluster computing using the criteria of structural complexity] / V.E. Podolskiy, S.S. Tolstyih // Vtoraya Sibirskaya shkola-seminar po parallelnyim vyichisleniyam. – Tomsk: Izd-vo Tom. un-ta, 2004. – S. 45-50. (In Russian)

Мультиагентные методы и инструментальные средства управления в сервис-ориентированной распределенной вычислительной среде

[1] *И.В. Бычков <bychkov@icc.ru>*
[1] *Г.А. Опарин <oparin@icc.ru>*
[1] *А.Г. Феоктистов <agf@icc.ru>*
[1] *В.Г. Богданова <bvg@icc.ru>*
[1] *А.А. Пашинин <apcrol@gmail.ru>*
[1] *Институт динамики систем и теории управления СО РАН, 664033, Россия, г. Иркутск, ул. Лермонтова, д. 134.*

Аннотация. В статье обсуждаются вопросы, связанные с обеспечением эффективного масштабирования потоков заданий, порождаемых проблемно-ориентированными распределенными вычислительными системами, в разнородных Grid с гибридными узлами. Рассматриваются мультиагентные методы и инструментальные средства нового поколения, обеспечивающие эффективное управление комбинированными потоками заданий масштабируемых сервис-ориентированных программных комплексов и балансировкой нагрузки вычислительных ресурсов исполнительной среды. Описывается высокоуровневый программный инструментарий для построения сервисов масштабируемых программных комплексов. Отличительной особенностью представленных методов и средств от известных является использование элементов экономической теории регулирования спроса и предложения ресурсов в согласованном мультиагентном управлении вычислениями для кластерной Grid с гибридными узлами, как на уровне Grid, так и на уровне приложений. Функции проблемно-ориентированной вычислительной среды (пользовательского приложения) оформляется в виде Grid-сервисов на основе применения технологии Web Services Resource Framework и шаблонов взаимодействия с локальными менеджерами ресурсов узлов Grid. Для создания программных агентов пользовательского приложения, предоставляющих эти функции, используется инструментальная среда High-performance computing Service-oriented Multiagent System Framework, разработанная авторами. В качестве примеров организации проблемно-ориентированной вычислительной среды (пользовательского приложения) с помощью рассматриваемых в статье методов и средств приводится ряд научных сервисов для экспериментальной Grid. Анализируются результаты эффективности их функционирования.

Ключевые слова: проблемно-ориентированные распределенные вычисления; мультиагентное управление; сервисы; инструментальные средства.

1. Введение

В настоящее время одним из фундаментальных и практически важных направлений исследований по организации проблемно-ориентированных распределенных вычислительных систем является обеспечение эффективного масштабирования потоков заданий, порождаемых этими системами, в разнородных Grid с гибридными узлами. Задание представляет собой спецификацию процесса решения задач, содержащую информацию о требуемых вычислительных ресурсах, исполняемых прикладных программах, входных/выходных данных, а также другие необходимые сведения. Поток заданий в целом обладает целым рядом свойств, таких как мощность потока, порядок поступления заданий, количество заданий, поступающих в один момент времени, характер взаимодействия заданий, однородность заданий, платформонезависимость приложений для выполнения заданий и другие важные характеристики.

Предполагается, что масштабируемая проблемно-ориентированная распределенная вычислительная система (пользовательское приложение) включает набор прикладных программ для параллельного решения задачи с помощью различных вычислительных единиц (например, ядер) гибридных узлов кластерной Grid и порождает комбинированный поток заданий, объединяющий задания для этих прикладных программ. При этом вычислительная нагрузка, связанная с решением задачи, распределяется между вычислительными единицами гибридных узлов кластерной Grid, а время выполнения заданий комбинированного потока уменьшается обратно пропорционально количеству используемых вычислительных единиц с учетом их производительности в составе конкретного узла Grid. Создание системы управления комбинированными потоками заданий для распределенной вычислительной системы является нетривиальной и весьма актуальной проблемой. Для успешного решения этой проблемы необходимо, чтобы пользовательские приложения такого рода включали возможности, во-первых, мониторинга состояния узлов Grid (их доступности, готовности, надежности, параметров очередей, статусов запущенных заданий и др.) и гибкого управления заданиями (учета требований к вычислительной системе, запуска, рестарта и миграции заданий, поддержки механизмов создания контрольных точек), во-вторых, динамической декомпозиции исходной задачи на подзадачи на основе анализа алгоритмов решения задачи и вычислительных характеристик узлов, назначения этих узлов для решения в них подзадач и последующей генерации потоков заданий для прикладных программ, размещенных в выбранных узлах.

Традиционные метапланировщики Grid, например, GridWay [1], предоставляют средства только для реализации в том или ином виде первой категории вышеперечисленных возможностей для пользовательских приложений. Реализация же второй категории возможностей требует

разработки таких средств управления, которые могли бы получать и применять знания о специфике проблемной области решаемой задачи.

Анализ мировых тенденций в области автоматизации решения прикладных задач в параллельных и распределенных вычислительных средах позволяет утверждать, что решение этой проблемы непосредственно связано с интеллектуализацией, так называемого, промежуточного программного обеспечения, позволяющего динамически интегрировать распределенные разнородные ресурсы в виртуальную исполнительную среду и предоставляющего возможности для прозрачного использования этой среды. Широко используемым на практике подходом к интеллектуализации промежуточного программного обеспечения является применение мультиагентных систем (МАС) для управления вычислениями [2]. Повышение качества управленческих решений в МАС зачастую достигается путем использования экономических механизмов регулирования спроса и предложения ресурсов распределенной вычислительной среды [3]. Можно выделить два основных подхода к мультиагентному управлению вычислениями [4]: взаимодействие МАС с локальными менеджерами ресурсов узлов Grid с целью оптимизации использования этих ресурсов и интеграция пользовательского приложения с МАС для выбора ресурсов с целью повышения эффективности решения задач этим приложением.

В первом случае, как правило, использование МАС предполагает замену традиционных метапланировщиков, таких как, например, GridWay, специальными агентами управления вычислениями, обеспечивающими более эффективное распределение ресурсов. Однако, вследствие такой замены, каждый пользователь, не зависимо от его желания, становится глобальным пользователем, осуществляющим взаимодействие с ресурсами распределенной вычислительной среды только с помощью МАС. Таким образом, ограничиваются возможности широкого круга локальных пользователей, желающих решать свои задачи в конкретных узлах среды, без использования промежуточного “посредника”. Кроме того, при управлении потоками заданий МАС на уровне Grid время выполнения отдельных приложений может увеличиваться, поскольку этим агентам не удается учесть ряд важных особенностей процесса решения задачи и пользовательских предпочтений, касающихся ресурсов. Во втором случае, при наличии большого числа приложений пользователей, использующих различные методы управления вычислениями, эффективность систем управления вычислениями может быть существенно снижена вследствие конкуренции агентов этих приложений за общие разделяемые ресурсы. Эта проблема во многом обуславливается тем, что сегодня нет более или менее известного стандартизированного инструментария, обеспечивающего построение системы управления для произвольной проблемно-ориентированной распределенной вычислительной системы.

В статье предлагаются мультиагентные методы и инструментальные средства организации проблемно-ориентированных вычислительных систем, обеспечивающие интеграцию двух рассмотренных выше подходов к управлению вычислениями в кластерной Grid. Предлагаемые методы и средства реализованы на основе парадигмы сервис-ориентированного программирования в рамках разрабатываемой в Институте динамики систем и теории управления (ИДСТУ) СО РАН САТУРН-технологии [5-7] построения интеллектуальных прикладных вычислительных систем и ее специализированных версий [8-10] для распределенных вычислительных сред.

2. Постановка задачи

Целью данной работы является разработка мультиагентных методов и инструментальных средств нового поколения, обеспечивающих эффективное управление комбинированными потоками заданий масштабируемых сервис-ориентированных программных комплексов и балансировкой нагрузки вычислительных ресурсов исполнительной среды. Эти разработки будут интегрированы в рамках единой технологии поддержки автоматизации процесса решения больших научных задач в современной кластерной Grid, узлы которой (кластеры) могут иметь сложную гибридную структуру. Исследование направлено на разработку высокоуровневых сервис-ориентированных информационно-вычислительных сред для прикладных специалистов, нацеленных на эффективное применение высокопроизводительных ресурсов без погружения в особенности низкоуровневого параллельного и/или распределенного программирования решаемой задачи. В таких средах построение параллельной (распределенной) крупноблочной программы на основе библиотеки специфицированных прикладных модулей и дальнейшее ее исполнение в распределенной среде производится автоматически по целевому содержательному запросу.

Для достижения поставленной цели представленного исследования необходимо было решить следующие задачи: создать высокоуровневый программный инструментарий для построения сервисов масштабируемых программных комплексов, обеспечивающий различные способы доступа к сервисам и возможность комплексного использования этих сервисов в процессе решения большой научной задачи в кластерной Grid; реализовать автоматизированную систему конвертации пользовательских запросов к масштабируемым сервис-ориентированным программным комплексам в комбинированные потоки заданий и распределения этих потоков в вычислительной среде; разработать мультиагентные методы и средства управления комбинированными потоками заданий в разнородной кластерной Grid, обеспечивающие реализацию многоуровневого параллелизма алгоритма решения задачи с учетом гибридной структуры узлов исполнительной среды.

В качестве апробации результатов исследования разработан ряд научных сервисов для экспериментальной кластерной Grid ИДСТУ СО РАН, проведены анализ и оценка показателей эффективности их функционирования в этой среде.

3. Методы организации распределенных вычислений

В общем случае пользовательское приложение должно быть представлено библиотекой прикладных программ, включающей, наряду с программами, реализующими алгоритмы решения прикладных задач (далее – решателями), специализированные программные модули, предназначенные для декомпозиции задачи по данным, а так же для ряда препроцессорных и постпроцессорных обработок входных и выходных данных. В частном случае для организации многовариантных расчетов применяются автоматизированные переборные методы формирования вариантов (сочетаний) значений входных переменных решаемой задачи на основе заданных областей допустимых значений переменных и шагов изменения этих значений или используются списки файлов с варианты значений входных переменных, заданные пользователями. Применение таких методов возлагается на систему управления вычислениями.

Управление распределенными вычислениями в кластерной Grid реализуется МАС [10] с заданной организационной структурой. Координация действий агентов осуществляется с помощью общих правил группового поведения. Агенты функционируют в соответствии с заданными ролями, и для каждой роли определены свои правила поведения в виртуальном сообществе агентов. МАС включает агентов постановки задачи, планирования вычислений, мониторинга и распределения ресурсов, классификации, конкретизации и выполнения заданий, а также управляющего агента. В разных виртуальных сообществах, возникающих в МАС, агенты могут координировать свои действия путем кооперации или соперничества. Агенты, представляющие пользовательское приложение, образуют виртуальное сообщество приложения (ВСП).

Распределение вычислительных ресурсов агентами базируется на использовании модели закрытого аукциона Викри [11]. По окончании торгов на таком аукционе агентами распределения ресурсов достигается согласованное устойчивое состояние, которое в определенной степени является аналогом равновесия по Нэшу в теоретико-игровых моделях [11].

Приложение пользователя оформляется в виде Grid-сервиса. К настоящему времени разработан широкий спектр инструментов для построения подобных сервисов [12]. Используемый в работе метод создания Grid-сервисов приложений базируется на сочетании технологий Web Services Resource Framework (WSRF) [13] и использовании шаблонов [14] взаимодействия с

локальными менеджерами ресурсов узлов Grid. Для создания ВСП используется инструментальная среда High-performance computing Service-oriented Multiagent System (HpcSoMaS) Framework, разработанная авторами на основе этих технологий.

4. Архитектура инструментальной среды HpcSoMas Framework

В состав инструментальной среды HpcSoMaS Framework (рис. 1) входят: средства создания агентов на базе нейронных сетей; библиотека разработки сервисов на основе стандарта REST (Representational State Transfer) и протокола SOAP (Simple Object Access Protocol), а так же готовые сервисы, реализующие базовые функции агентов ВСП, созданные на основе библиотечных классов и требующие для своего использования только конфигурационную настройку; графические средства проектирования сервисов; документация по используемым форматам файлов конфигурации сервисов. Использование стандарта REST обусловлено его возможностями для представления сервиса в виде клиент-серверного приложения, выполнения нересурсоемких работ в фоновом режиме, запуска потока на каждого клиента, компактностью пакетов запросов и ответов по сравнению с SOAP.

Существуют две категории пользователей рассматриваемой инструментальной среды: системные разработчики, квалификация которых позволяет модифицировать базовые возможности готовых сервисов при создании системной части программных комплексов, и специалисты-предметники, занимающиеся созданием функционального наполнения приложения и использующие специальные инструменты для его представления в виде сервиса. Для первой категории пользователей предоставляется режим работы с библиотекой разработки сервисов, предусматривающий ручное заполнение файлов конфигурации сервисов, доработку и компиляцию исходного программного кода шаблонов сервисов. Для второй категории пользователей предоставляется стандартный набор сервисов, для которых файлы настроек можно создавать и модифицировать, используя утилиты с графическим интерфейсом.

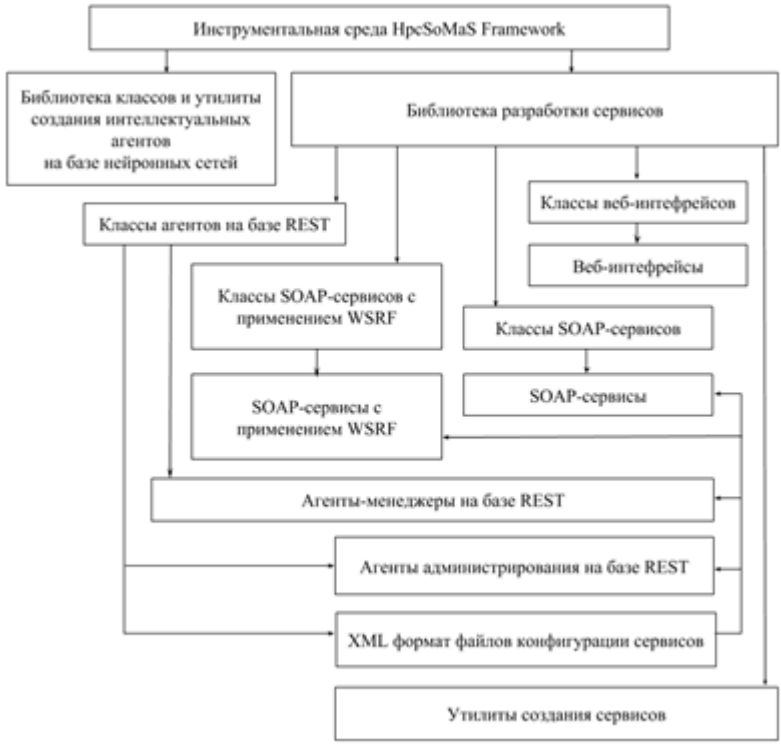


Рис.1. Инструментальная среда создания ВСП агентов.

В ВСП, разрабатываемых с помощью инструментальной среды HpcSoMaS Framework, можно выделить три уровня: уровень пользовательских агентов, представленный клиентами в браузере; уровень агентов-менеджеров, представленный сервисами, реализованными на основе стандарта REST; уровень реактивных агентов выполнения заданий, представленный сервисами, реализованными на основе протокола SOAP. Многоуровневая архитектура создаваемых ВСП позволяет локально заменять отдельные функциональные части этого сообщества и многократно их использовать в разных приложениях. Пользователи могут подключаться к любому из агентов-менеджеров через web-интерфейс пользовательского агента, доступный с компьютеров и мобильных устройств, подключенных к сети Интернет, при наличии учетной записи пользователя на ВК агента.

5. Вычислительный эксперимент

Сервисы пользовательских приложений были созданы для использования в разнородной кластерной Grid ИДСТУ СО РАН, включающей: однородный ВК «Blackford», 20 узлов, 40 CPU Quad-Core Intel Xeon E5345, общее число ядер 160; неоднородный ВК «Академик В.М. Матросов», 110 узлов, 220 CPU AMD Opteron 6276 «Bulldozer», общее число ядер 3520, узел с графическими процессорами NVidia C2070 («Fermi»); неоднородный ВК с GPU NVidia «Tesla», в состав которого входит 4 четырехъядерных процессора Intel Xeon X5570 («Nehalem») и 8 GPU NVidia «Tesla» C1060 с общим числом потоковых ядер 1920; ВК ПЭВМ, 8 рабочих станций, 1 CPU AMD, общее число ядер 32. На каждом ВК был установлен HTTP web-сервер, обеспечивающий доступ извне, для REST- и SOAP-сервисов использовался сервер TomEE, который представляет из себя модифицированный Apache Tomcat с добавленным в него функционалом JavaEE. На рис. 2. приведена схема размещения и функционирования ВСП в кластерной Grid ИДСТУ СО РАН.

В качестве первого примера был реализован сервис для решения задачи построения области устойчивости в пространстве двух выбранных параметров K и T регулятора замкнутой системы управления, описываемой дифференциальным уравнением $\frac{dX}{dt} = Ax$, где элементы матрицы A зависят от параметров K и T. Эта задача сводится к решению множества независимых подзадач (проведению многовариантных расчетов) по определению устойчивости матрицы A при изменении значений параметров K и T в заданных диапазонах $K_{\min} \le K \le K_{\max}$ и $T_{\min} \le T \le T_{\max}$ с шагом ΔK и ΔT соответственно. Путем варьирования значений параметров K и T строится числовая сетка, на основе которой формируется множество подзадач. Для выполнения задания локальный агент запускает приложение, реализующее схему вычисления собственных значений произвольной плотной матрицы с использованием алгоритмов, представленных в работе [15]. Необходимым условием решения исходной задачи является выполнение задания для решения каждой подзадачи.

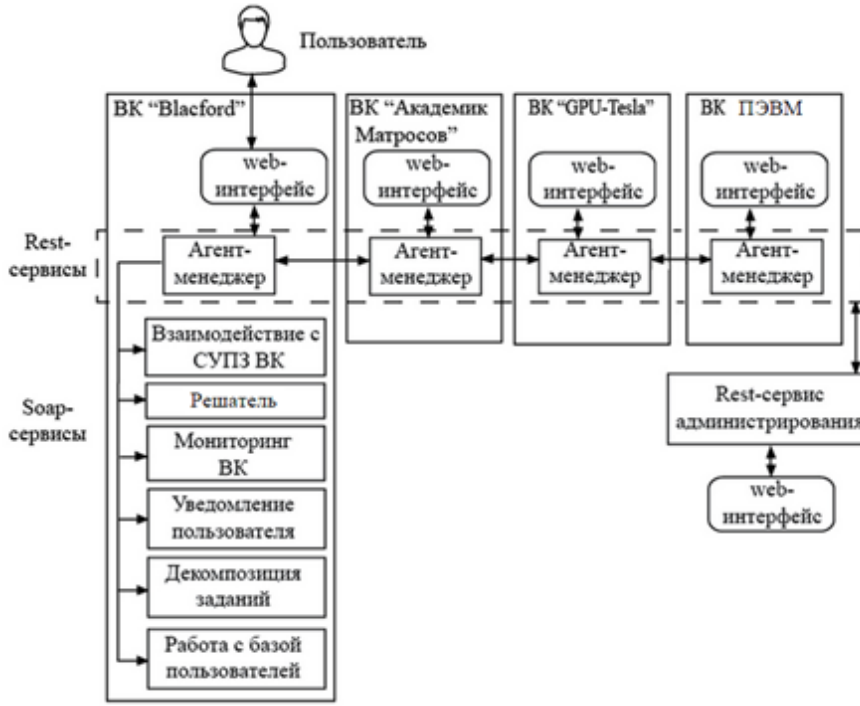


Рис.2. Схема размещения и функционирования ВСП в кластерной Grid ИДСТУ СО РАН.

В качестве второго примера реализован сервис для решения SAT-задач (задач выполнимости булевых ограничений) путем проведения многовариантных расчетов на основе декомпозиции исходной задачи с помощью метода расщепления [16]. В этом случае многовариантные расчеты могут выполняться в режиме динамического выбора ресурсов. В вычислительных экспериментах использовались как существующие SAT-решатели с открытой лицензией, так и разработанные авторами решатели [17, 18].

В первом примере сервис приложения предназначен для реализации многовариантных расчетов с использованием статического выбора ресурсов. Очевидно, что для задач такого вида преимущество рассмотренных выше средств мультиагентного управления по сравнению с традиционными менеджерами ресурсов незначительно. Во втором же примере сервис приложения предназначен для реализации многовариантных расчетов в режиме динамического выбора ресурсов. В этом случае, предлагаемые средства мультиагентного управления вычислениями позволяют существенно сократить время решения задачи по сравнению с традиционными менеджерами ресурсов за счет выбора наиболее оптимальных решателей и

наименее загруженных вычислительных ресурсов, соответствующих этим решателям (табл. 1). В таблице приведены результаты решения задачи Эйлера о ходе шахматного коня (незамкнутое движение) под управлением традиционного менеджера ресурсов и под управлением ВСП, обеспечивающего возможность динамического перераспределения подзадач при появлении свободных ресурсов. Приводится среднее и минимальное время для серии из 100 запусков задач.

Табл.1. Сравнение времени решения sat-задачи при распределении потока заданий традиционным менеджером ресурсов (МР) и ВСП.

		Среднее время решения sat-задачи, с.		Минимальное время решение sat-задачи, с.	
КНФ	Число переменных/ дизъюнктов	МР	ВСП	МР	ВСП
knight8	4096/491024	183.0	132.0	61.0	0.6
knight9	6561/1007603	499.0	359.0	282.0	199.0
knight10	10000/1913276	3599.0	2464.0	651.0	276.0

Третий сервис предназначен для организации многовариантных расчетов при исследовании систем массового обслуживания в среде имитационного моделирования GPSS World [19]. Необходимость проведения таких исследований периодически возникает, например, в региональных складских логистических комплексах. В числе решаемых задач: моделирование погрузочно-разгрузочных работ, моделирование технического обслуживания спецтехники и автотранспорта, построение расписаний обслуживания плановых клиентов с учетом потоков случайных заявок на обслуживание, разработка моделей обслуживания клиентов с различными уровнями обслуживания, прогнозирование процессов сдачи складских объектов в аренду, анализ фондоемкости торгово-складских услуг. Решение перечисленных выше задач требует проведения многовариантных расчетов. Учитывая стохастичность моделируемых процессов, для каждого варианта нужно выполнить достаточно большое число прогонов с тем, чтобы добиться требуемой степени достоверности результатов. Дальнейшее развитие сервиса предполагает расширение его функций функциями планирования вычислительного эксперимента, анализа и интерпретации результатов моделирования.

6. Имитационное моделирование функционирования Grid

С целью более полного исследования рассмотренных выше методов и инструментальных средств проведено имитационное моделирование функционирования Grid с помощью системы GPSS World. Моделируемая система включала 10 кластеров с числом ядер от 6000 до 14000 единиц и 300 пользователей. Общее число ядер составляло 100000 единиц. Число доступных ядер в экспериментах изменялось от 80000 до 100000. Выбранная конфигурация моделируемой вычислительной среды по своим параметрам достаточно адекватно отражает характеристики российских суперкомпьютеров, входящих в официальный список 500 мощнейших компьютеров мира [20]. Кластеры включали гибридные узлы, поддерживающие различные технологии параллельного программирования. При проведении имитационного моделирования предполагалось, что задания пользователей на уровне Grid могут быть выполнены на любом кластере, а для каждого приложения локального пользователя кластеров (ЛПК) имеются уникальные решатели, реализованные с учетом вычислительных особенностей узлов кластеров, в которых эти решатели установлены. При имитации времени выполнения задания на кластерах применялись коэффициенты ускорения счета, значения которых для разных кластеров варьировались от 1 до 1.5 в зависимости от вычислительных характеристик этих кластеров.

Моделируемый период времени работы системы – 30 суток. За этот период обработано 12990 потоков заданий как пользователей на ровне Grid, так и локальных пользователей кластеров. Эти потоки включали от 1000 до 10000 процессов для параллельных программ или заданий для многовариантных расчетов. В качестве метапланировщиков Grid использовались GridWay или MAC [10], а распределение ресурсов для приложений локальных пользователей осуществлялось самими пользователями или ВСП. Интервал перераспределения ресурсов (RESCHEDULING_INTERVAL) как для GridWay, так и для ВСП был равен 5 мин. Дисциплина обслуживания очередей заданий – FCFS (First Come, First Served) с приоритетами. В качестве основных наблюдаемых переменных имитационной модели использованы следующие показатели: среднее число n_{avg} заданий в очереди кластера, среднее время t_{avg} пребывания задания в очереди кластера и средний коэффициент k_{avg} полезного использования узлов кластеров, среднеквадратическое отклонение σ коэффициент полезного использования узлов кластеров.

Таблица 2. Результаты моделирования для 100000 доступных ядер.

Показатель	Менеджер ресурсов: уровень Grid, уровень приложения		
	GridWay, ЛПК	MAC, ЛПК	MAC, ВСП
n_{avg}, единиц	520.362	98.071	53.859
t_{avg}, сек.	701.750	286.184	170.183
k_{avg}, %	0.690	0.616	0.647
σ	0.005	0.003	0.002

Результаты моделирования (табл. 2), показывают, что применение ВСП совместно с MAC может существенно улучшить выбранные показатели функционирования достаточно большой кластерной Grid по сравнению как с метапланировщиком GridWay, так и с самой MAC. Во многом это обуславливается следующим обстоятельством: когда вместо миграции задания по выполнению программы с одного кластера на другой требуется генерация нового задания для выполнения другой программы, метапланировщики на уровне Grid не могут выполнять подобные действия, так как не обладают всей необходимой информацией о проблемной области решаемой задачи пользователя. При разработке ВСП пользователь имеет возможность снабдить агентов нужными знаниями. Используемый в MAC и ВСП алгоритм распределения ресурсов на основе экономических механизмов регулирования их спроса и предложения позволяет осуществлять положительное влияние на балансировку загрузки ресурсов, о чем свидетельствует уменьшение показателя σ. Уменьшение значения среднего коэффициента k_{avg} полезного использования узлов кластеров в третьем и четвертом столбцах табл. 2 по сравнению со значением во втором столбце объясняется сокращением общего времени выполнения всех потоков заданий за счет учета агентами MAC и ВСП коэффициентов ускорения счета на кластерах при распределении в них заданий и тем самым увеличения числа заданий с нулевым временем ожидания в очереди.

7. Заключение

В статье рассмотрены новые мультиагентные методы и инструментальные средства управления комбинированными потоками заданий в разнородной кластерной Grid, обеспечивающие реализацию многоуровневого параллелизма алгоритма решения задачи с учетом гибридной структуры узлов исполнительной среды. Отличительными особенностями этих методов и средств от известных являются: использование элементов экономической теории регулирования спроса и предложения ресурсов в согласованном мультиагентном управлении вычислениями для Grid с гибридными узлами, как на уровне Grid, так и на уровне приложений; реализация агентов ВСП в

виде сервисов. Также следует отметить, что при взаимодействии с ресурсами Grid агенты на обоих уровнях используют команды локальных менеджеров ресурсов, доступные любому пользователю, запускают задания через общие очереди кластеров Grid и не влияют на их административные политики.

Список литературы

[1]. Herrera J., Huedo E., Montero R., Llorente I. Porting of Scientific Applications to Grid Computing on GridWay // Scientific Programming. 2005. Vol. 13. No. 4. P. 317–331.

[2]. Durfee E.H. Distributed Problem Solving and Planning // Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence / Ed. by G. Weiss. MIT Press, 1999. P. 121-164.

[3]. Market-Oriented Grid and Utility Computing / Ed. by R. Buyya, K. Bubendorfer. Wiley & Sons, 2010. 643 p.

[4]. Топорков В.В. Модели распределенных вычислений. М.: Физматлит, 2004. 320 с.

[5]. Опарин Г.А. Сатурн – метасистема для построения пакетов прикладных программ // Пакеты прикладных программ. Методы и разработки. Новосибирск: Наука, 1982. С. 130–160.

[6]. Лавров С., Левин Д., Матулис В., Мацкин М., Нариньяни А., Опарин Г., Тыугу Э., Хорошевский В. Технологические системы поддержки разработок искусственного интеллекта // Представление знаний в человеко-машинных и робототехнических системах. М.: ВЦ АН СССР; ВИНИТИ, 1984. Т. В. С. 102–123.

[7]. Опарин Г.А., Феоктистов А.Г., Феоктистов Д.Г., Журавлев А.Е. Инструментальные средства построения и эксплуатации пакетов знаний // Управляющие системы и машины. 1997. № 1-3. С. 138-143.

[8]. Опарин Г.А., Феоктистов А.Г. Инструментальная распределенная вычислительная САТУРН-среда // Программные продукты и системы. 2002. № 2. С. 27–30.

[9]. Васильев С.Н., Опарин Г.А., Феоктистов А.Г., Сидоров И.А. Интеллектные технологии и инструментальные средства создания вычислительной инфраструктуры в сети Интернет // Вычислительные технологии. 2006. Т. 11. № S8. С. 34-44.

[10]. Bogdanova V.G., Bychkov I.V., Korsukov A.S., Oparin G.A., Feoktistov A.G. Multiagent Approach to Controlling Distributed Computing in a Cluster Grid System // Journal of Computer and Systems Sciences International. 2014. Vol. 53. No. 5. Pp. 713–722. DOI: 10.1134/S1064230714040030.

[11]. Николенко С.И. Теория экономических механизмов. М.: Интуит.ру; Бином. Лаборатория знаний, 2009. 207 с.

[12]. Buyya R., Vecchiola C., Selvi S.T. Mastering Cloud Computing. Burlington, Massachusetts, USA: Morgan Kaufmann, 2013. 469 p.

[13]. Czajkowski K., Ferguson D.F., Foster I., Frey J., Graham S., Sedukhin I., Snelling D., Tuecke S., Vambenepe W. The WS-Resource Framework. Version 1.0. http://www.globus.org/wsrf/specs/ws-wsrf.pdf.

[14]. Бычков И.В., Опарин Г.А., Феоктистов А.Г., Богданова В.Г., Корсуков А.С. Сервис-ориентированный подход к организации распределенных вычислений с помощью инструментального комплекса DISCENT // Информационные технологии и вычислительные системы. 2014. № 2. С. 7-15.

[15]. Wilkinson J.X., Reinsch C. Handbook for Automatic Computation. Volume II: Linear Algebra. Shpringer-Verlag. 1971. 448 p.

[16]. Oparin G.A., Feoktistov A.G., Bogdanova V.G., Novopashin A.P. The solution of Boolean equations of high dimensionality in the distributed computing environment // Distributed Computing and Grid-Technologies in Science and Education: Book of the Abstr. of the Intern. Conf. Dubna, Russia, June 29 -July 2, 2004. JINR, 2004. D11-2004-82. P. 65.
[17]. Опарин Г.А., Богданова В.Г. РЕБУС – интеллектуальный решатель комбинаторных задач в булевых ограничениях // Вестник НГУ. Серия: Информационные технологии. 2008. Т. 6. Вып. 1. С. 60-68.
[18]. Опарин Г.А., Богданова В.Г. Инструментальные средства автоматизации параллельного решения булевых уравнений на многоядерных процессорах // Программные продукты и системы. 2012. № 1. С. 10-14.
[19]. Боев В.Д. Моделирование систем. Инструментальные средства GPSS World. СПб.: БХВ-Петербург. 2004. 368 с.
[20]. TOP500 Supercomputing Sites // Prometeus GmbH. Режим доступа: http://www.top500.org (дата обращения 01.12.2014).

Multiagent methods and tools of management in a service-oriented distributed computing environment

[1] *I. Bychkov < bychkov@icc.ru >*
[1] *G. Oparin <oparin@icc.ru>*
[1] *A. Feoktistov <agf@icc.ru>*
[1] *V.Bogdanova <bvg@icc.ru>*
[1] *A. Pashinin <apcrol@gmail.ru>*
[1] *Institute for System Dynamics and Control Theory of Siberian Branch of Russian Academy of Sciences*
134, Lermontova st., Irkutsk, 664033, Russia.

Abstract. The paper discusses the issues related to ensuring the effective scaling of the job flow generated by problem-oriented distributed computing systems in heterogeneous Grid with hybrid nodes. Considers the multiagent methods and tools of the new generation, to ensure effective management of the combined job flow of scalable service-oriented software systems and load balancing for computing resources of runtime environment. Describes a high-level tools for developing services of scalable software systems. A distinctive feature of the presented methods and tools is use of elements of the economic theory (such as elements of regulation of supply and demand of resources) in the coordinated multiagent management of cluster Grid with hybrid nodes, both at the Grid level, and at the application level. The functions of problem-oriented computing environment (user application) is realized in the form of Grid-services using technology Web Services Resource Framework and patterns of interaction with local resource managers of Grid nodes. To create the agent of application is used development environment High-performance computing Service-oriented Multiagent System Framework, developed by the authors. A number of scientific services for experimental Grid are given as examples of organization of the problem-oriented computing environment (user application) using methods and tools considered in this paper. The results of their effective functioning is analyzed.

Keywords: problem-oriented distributed computing; multiagent control; services; tools.

References

[1]. Herrera J., Huedo E., Montero R., Llorente I. Porting of Scientific Applications to Grid Computing on GridWay. Scientific Programming. 2005, vol. 13, no. 4, pp. 317–331.
[21]. Durfee E.H. Distributed Problem Solving and Planning. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence / Ed. by G. Weiss. MIT Press, 1999, pp. 121-164.
[22]. Market-Oriented Grid and Utility Computing / Ed. by R. Buyya, K. Bubendorfer. Wiley & Sons, 2010. 643 p.
[23]. Toporkov V.V. Modeli raspredelennykh vychislenij [Models of distributed computing] M.: Fizmatlit [Fizmatlit], 2004. 320 p. (In Russian).

[24]. Oparin G.A. Saturn – metasistema dlya postroeniya paketov prikladnykh programm [Metasystem for developing applied software packages]. Pakety prikladnykh programm. Metody i razrabotki [Applied software packages. Methods and developments]. Novosibirsk: Nauka [Science], 1982, pp. 130–160. (In Russian).

[25]. Lavrov S., Levin D., Matsulis V., Matskin M., Narin'yani A., Oparin G., Tyugu Je., Khoroshevskij V. Tekhnologicheskie sistemy podderzhki razrabotok iskusstvennogo intellekta [Technological systems for support of development of artificial intelligence]. Predstavlenie znanij v cheloveko-mashinnykh i robototekhnicheskikh sistemakh [Knowledge representation in human-machine and robot-technical systems]. M.: VC AN SSSR; VINITI [Moskow, Computing center of USSR Academy of Sciences, Viniti]. 1984, vol. B, pp. 102–123. (In Russian).

[26]. Oparin G.A., Feoktistov A.G., Feoktistov D.G., Zhuravlev A.E. Instrumental'nye sredstva postroenija i ekspluatatsii paketov znanij [Tools for developing and operating of knowledge packages]. Upravlyajushhie sistemy i mashiny [Control systems and machines]. 1997, no 1-3, pp. 138-143. (In Russian).

[27]. Oparin G.A., Feoktistov A.G. Instrumental'naya raspredelennaya vychislitel'naya SATURN-sreda [Distributed computing framework SATURN]. Programmnye produkty i sistemy [Software & Systems]. 2002, no. 2, pp. 27–30. (In Russian).

[28]. Vassilyev S.N., Oparin G.A., Feoktistov A.G., Sidorov I.A. Intellektnye tekhnologii i instrumental'nye sredstva sozdaniya vychislitel'noj infrastruktury v seti Internet [Intelligent technology and tools for developing of the computational infrastructure in Internet]. Vychislitel'nye tehnologii [Computational Technologies]. 2006, vol. 11, no S8, pp. 34-44. (In Russian).

[29]. Bogdanova V.G., Bychkov I.V., Korsukov A.S., Oparin G.A., Feoktistov A.G. Multiagent Approach to Controlling Distributed Computing in a Cluster Grid System. Journal of Computer and Systems Sciences International. 2014. Vol. 53. No. 5. Pp. 713–722. DOI: 10.1134/S1064230714040030.

[30]. Nikolenko S.I. Teoriya ekonomicheskih mekhanizmov [Theory of economic mechanisms] M.: Intuit.ru; Binom. Laboratoriya znanij [Binom. Knowledge lab], 2009. 207 p. (In Russian).

[31]. Buyya R., Vecchiola C., Selvi S.T. Mastering Cloud Computing. Burlington, Massachusetts, USA: Morgan Kaufmann, 2013. 469 p.

[32]. Czajkowski K., Ferguson D.F., Foster I., Frey J., Graham S., Sedukhin I., Snelling D., Tuecke S., Vambenepe W. The WS-Resource Framework. Version 1.0. www.globus.org/wsrf/specs/ws-wsrf.pdf.

[33]. Bychkov I.V., Oparin G.A., Feoktistov A.G., Bogdanova V.G., Korsukov A.S. Servis-orientirovannyj podkhod k organizatsii raspredelennykh vychislenij s pomoshh'ju instrumental'nogo kompleksa DISCENT [The service-oriented approach to distributed computing on the basis of the toolkit DISCENT]. Informatsionnye tekhnologii i vychislitel'nye sistemy [Information technology and computer systems]. 2014, no. 2, pp. 7-15. (In Russian).

[34]. Wilkinson J.X., Reinsch C. Handbook for Automatic Computation. Volume II: Linear Algebra. Shpringer-Verlag. 1971. 448 p.

[35]. Oparin G.A., Feoktistov A.G., Bogdanova V.G., Novopashin A.P. The solution of Boolean equations of high dimensionality in the distributed computing environment . Distributed Computing and Grid-Technologies in Science and Education: Book of the Abstr. of the Intern. Conf. Dubna, Russia, June 29 -July 2, 2004. JINR, 2004. D11-2004-82. P. 65.

[36]. Oparin G.A., Bogdanova V.G. REBUS – intellektual'nyj reshatel' kombinatornykh zadach v bulevykh ogranicheniyah [Intelligent solver of combinatorial problems in boolean constraints]. Vestnik NGU. Serija: Informatsionnye tekhnologii [Bulletin of NGU. Series: Information technology]. 2008, vol. 6, no. 1, pp. 60-68. (In Russian).
[37]. Oparin G.A., Bogdanova V.G. Instrumental'nye sredstva avtomatizatsii parallel'nogo resheniya bulevykh uravnenij na mnogoyadernyh processorakh [Parallel computing toolkit for solving Boolean equations on multi-core processors]. Programmnye produkty i sistemy [Software & Systems]. 2012, no. 1, pp. 10-14. (In Russian).
[38]. Boev V.D. Modelirovanie sistem. Instrumental'nye sredstva GPSS World [Simulation systems. GPSS World tools] SPb.: BHV-Peterburg [BHV-Peterburg]. 2004. 368 p. (In Russian).
[39]. TOP500 Supercomputing Site. www.top500.org

Improved known plaintexts attack on Domingo-Ferrer homomorphic cryptosystem[1]

A.V. Trepacheva <alina1989malina@ya.ru>
Southern Federal University,
105/42, Bolshaya Sadovaya st., Rostov-on-Don, 344006, Russia.

Abstract. This paper is devoted to known plaintexts cryptanalysis of homomorphic cryptosystem proposed by Domingo-Ferrer. In previous works it was shown that at least $d+1$ pairs (plaintext, ciphertext) are necessary to recover secret key, where d is a degree of polynomials representing ciphertexts. Here we analyze existing known plaintext attack. And also slightly modified attack on this cryptosystem is presented. It allows to reduce the necessary number of pairs meaningfully. In particular interception only of two pairs may be enough for successful key recovering with overwhelming probability. The running time of our attack depends polynomially on d and logarithmically on plaintexts space size as well as for previous attack. We provide the results of computer experiments.

Key words: known plaintext cryptanalysis; homomorphic encryption; cloud computations.

1. Introduction

Homomorphic encryption (HE) is a cryptographic primitive supporting the additional property in comparison with ordinary encryption: *HE allows computing over encrypted data.* Let's explain what this means. We assume that plaintexts space P and ciphertexts space C are rings with operations $+_P, \cdot_P$ and $+_C, \cdot_C$ correspondingly. And let E, D be encryption and decryption functions of cryptosystem ε. The last one is homomorphic if for $\forall x, y \in P$ and $\forall E(x), E(y) \in C$ the following properties are satisfied:

$$D(E(x) +_C E(y)) = x +_P y, \quad (1)$$

$$D(E(x) \cdot_C E(y)) = x \cdot_P y. \quad (2)$$

So the result of computations over ciphertexts will be an encryption of computations result over underlying plaintexts.

[1] This work is supported by grant RFBR 15-07-00597-a

Homomorphic cryptosystems (HC) are of key importance for protecting sensitive data in clouds. Computationally weak clients may outsource computations over their data while keeping this data in secret. This makes the development of new homomorphic cryptosystems and cryptanalysis of existing a hot topic.

By the present moment a variety of homomorphic cryptosystems were proposed (for example see [1-5]). RSA [1] is one of the most well known, because the product of RSA ciphertexts is an encryption of corresponding plaintexts product. But cryptosystems [1-5] are partially homomorphic, because they allow to compute over ciphertexts only functions lying in some bounded class. In particular for [1] only property (2) holds (multiplicatively homomorphic cryptosystems). Whereas for instance for [2] only (1) holds (additively homomorphic).

The simplest example of HC holding both (1), (2) was introduced in the fundamental paper [6] of Rivest, Adleman and Dertouzos. Encryption function $E: \mathbf{Z}_n \to \mathbf{Z}_p \times \mathbf{Z}_q$ works as follows $x \in \mathbf{Z}_n \to (x \bmod p, x \bmod q)$. Unfortunately, in [7] such encryption was shown to be unsecure against known plaintext attack (KPA). Beginning with [6] lots of cryptosystems with properties (1), (2) were suggested. Here two the most important groups may be highlighted. In the first group there are cryptosystems [8-11] with unlimited ciphertexts sizes growth during computing over them (their security analysis may be founded in [12,13]). Whereas cryptosystems of second group have some polynomially bounds on ciphertexts sizes growth. In this group for example there are cryptosystems [14-18] belonging to direction initiated by innovative work [14] of IBM researcher Craig Gentry.

Second group obviously is more interesting for practice. But unfortunately existing cryptosystems are not enough efficient for usage in real applications. The development of Gentry-like HCs now has mostly theoretical character. And in practice at the present moment HCs from the first group are used. For instance cryptosystems [10, 11] proposed by Domingo-Ferrer are exploited in secure packet forwarding in mobile ad hoc networks (see [19-24]). The main reason is a conceptual simplicity of constructions from [10, 11].

In the light of this the analysis of Domingo-Ferrer HCs resistance to different attacks is of value. Here we will concentrate on KPA. In [25] the authors described KPA on [10] and showed that to recover secret key an adversary A should intercept $t \geq d+1$ pairs (plaintext, ciphertext), where d is a degree of polynomials representing ciphertext. The aim of the present work to demonstrate that [10] may be broken using even two pairs (plaintext, ciphertext). We give some theoretical reasoning to this fact. And also we provide an experimental confirmation.

2. Denotations

All logarithms are base-2. A probability of event M is denoted by $\Pr(M)$, ring of integers – by $\mathbf{Z}$, ring of integers modulo n – by $\mathbf{Z}_n$, the multiplicative subgroup of $\mathbf{Z}_n$ – by $\mathbf{Z}_n^*$. An adversary trying to break cryptosystem will be denoted by A. For

symmetric cryptosystem ε: P – plaintexts space, C – ciphertexts space, K – secret keys space, $\mathbf{D}$ – probabilistic distribution over P.

We denote by $x \xleftarrow{\$} R$ a random element sampled according to uniform distribution over ring R and also by $x \xleftarrow{\mathbf{D}} R$ – random ring element generated according distribution $\mathbf{D}$ over R. Denotation $f(x) \xleftarrow{\$} R[x]$ means that all coefficients of polynomial f are random values chosen uniformly and independently from R.

3. Overview of Domingo-Ferrer cryptosystem

Let's briefly recall cryptosystem from [10]. The author sets $P = \mathbf{Z}_n$, $C \subset \mathbf{Z}_p[x] \times \mathbf{Z}_q[x]$, $K = \mathbf{Z}_p^* \times \mathbf{Z}_q^*$, where $n = p \cdot q$, $p, q -$ big primes, $p < q$, $\log p \approx \log q$, i.e. n – RSA modulus. Its factorization is a secret. Secret key is a pair $k = (r_p, r_q) \in K$. Before encryption public parameter $d \in \mathbf{Z}_+$ is fixed.

Encryption($a \in \mathbf{Z}_n, d \in \mathbf{Z}_+, p, q, k = (r_p, r_q) \in K$ **):**

- $a \in \mathbf{Z}_n \to a'(x) \in \mathbf{Z}_n[x]$, where $a'(x) = \sum_{i=1}^{d} a'_i \cdot x^i$ and for $i = \overline{2, d-1}: a'_i \xleftarrow{\$} \mathbf{Z}_n$, $a'_d \xleftarrow{\$} \mathbf{Z}_n \setminus \{0\}$ and $a'_1 := (a - \sum_{i=1}^{d-1} a'_i) \bmod n$.
- Ciphertext is a pair of polynomials $c = (c_p(x), c_q(x))$, where $c_p(x) := a'(r_p \cdot x) \bmod p$ and $c_q(x) := a'(r_q \cdot x) \bmod q$.

One may see that $a \equiv a'(1) (\bmod n)$ (or $a \equiv \sum_{i=1}^{d} a'_i (\bmod n)$).

Decryption($c = (c_p(x), c_q(x))$, p, q, $k^{-1} = (r_p^{-1}, r_q^{-1})$ **):**

- $a'_p(x) := c_p(r_p^{-1} \cdot x) \bmod p$ $a'_q(x) := c_q(r_q^{-1} \cdot x) \bmod q$ (clear $a'_p(x) \equiv a'(x) (\bmod p)$ and $a'_q(x) \equiv a'(x) (\bmod q)$).
- $a_p := a'_p(1) \bmod p$, $a_q := a'_q(1) \bmod q$ (clear $a \equiv a_p (\bmod p)$, $a \equiv a_q (\bmod q)$).
- $a := CRT(a_p, a_q, p, q)$, where $CRT(a_p, a_q, p, q)$ means the reconstruction of $a \in \mathbf{Z}_n$ by $a_p \in \mathbf{Z}_p$, $a_q \in \mathbf{Z}_q$ using Chinese reminder theorem.

In [10] the author suggested two regimes of cryptosystem working. In the first variant modulus n is public and plaintexts and ciphertexts coefficients are treated by untrusted party as elements of $\mathbf{Z}_n$. In the second case n is hidden for providing

higher level of security. And then plaintexts and ciphertexts coefficients are treated as elements of $\mathbf{Z}$. Here we will consider only the first case.

Homomorphic properties: Let's suppose there are plaintexts $a_1, a_2 \in \mathbf{Z}_n$ and $c_1 = (c_{p,1}(x), c_{q,1}(x))$, $c_2 = (c_{p,2}(x), c_{q,2}(x))$ – its encryptions made on the same key $k = (r_p, r_q)$ and for the same d. In [10] the author proves the following statements.

Statement 1. Ciphertext $c_+ = ((c_{p,1}(x) + c_{p,2}(x)) \bmod n, (c_{q,1}(x) + c_{q,2}(x)) \bmod n)$ is a correct encryption of plaintext $(a_1 + a_2) \bmod n \in \mathbf{Z}_n$ for key $k = (r_p, r_q)$ and parameter d.

Statement 2. Ciphertext $c_* = ((c_{p,1}(x) \cdot c_{p,2}(x)) \bmod n, (c_{q,1}(x) \cdot c_{q,2}(x)) \bmod n)$ is a correct encryption of plaintext $(a_1 \cdot a_2) \bmod n \in \mathbf{Z}_n$ for key $k = (r_p, r_q)$ and parameter $2 \cdot d$.

One may see that multiplication of ciphertexts causes an unbounded growth of their sizes (the size is doubled). So in general this HC isn't good for practice. But its simplicity makes it good for applications requiring only computations of some special functions (see [19-24]).

Remark 1. In practice for example $\log n \approx 2048$ may be chosen. Then the size S of ciphertext is $2048 \cdot d$ bits. This implies that $d \le 500$ should be chosen to obtain $S \le 10^6$ bits. Such setting seems reasonable because in all latest HCs [14-18] S is usually about 10^6 bits. Larger value of S will make homomorphic computations too much expensive. But of course it is suitable only if additive homomorphism is necessary. But if multiplicative homomorphism will be exploited then d should be smaller.

4. Cryptanalysis of Domingo-Ferrer cryptosystem

4.1 Existing KPA

Here we briefly discuss existing results [25] concerning known plaintexts analysis of Domingo-Ferrer cryptosystem [10]. Let's suppose A has t pairs $(a_i \in P, c_i \in C), i = \overline{1,t}$, where c_i is an encryption of a_i and all c_i are produced for the same n, $k = (r_p, r_q)$ and d. Ciphertexts c_i are pairs $(c_{p,i}(x) \in \mathbf{Z}_p[x],\ c_{q,i}(x) \in \mathbf{Z}_q[x])$, where $c_{p,i}(x) = \sum_{j=1}^{d} c_{p,i,j} \cdot x^j$, $c_{q,i}(x) = \sum_{j=1}^{d} c_{q,i,j} \cdot x^j$.

A needs to recover p, q, $k^{-1} = (r_p^{-1}, r_q^{-1})$ using n and $(a_i \in P, c_i \in C), i = \overline{1,t}$.

Remark 2. Here we consider the case of public n. So before recovering p, q A works with polynomials $c_{p,i}(x)$, $c_{q,i}(x)$ modulo n. In [25] the authors also propose

an attack for hidden n. And in this case coefficients $c_{p,i,j}$, $c_{q,i,j}$ are treated as integers at the first step of KPA.

According to encryption procedure the following congruences holds:

$$c_{p,i}(r_p^{-1}) - a_i \equiv 0 (\mathrm{mod}\, p), \quad (3)$$

$$c_{q,i}(r_q^{-1}) - a_i \equiv 0 (\mathrm{mod}\, q). \quad (4)$$

So polynomials $f_i(x) = c_{p,i}(x) - a_i \in \mathbf{Z}_n[x], i = \overline{1,t}$ have a common root r_p^{-1} modulo p. Similarly $g_i(x) = c_{q,i}(x) - a_i \in \mathbf{Z}_n[x], i = \overline{1,t}$ have a common root r_q^{-1} modulo q. And please note that r_p^{-1}, r_q^{-1} are not obligatory roots of $f_i(x)$, $g_i(x)$ modulo n. So KPA should proceed in three steps:

- A recovers secret modulus p and sets $q = n / p$.
- A computes r_p^{-1} as a common root of $f_i(x), i = \overline{1,t}$ modulo p.
- A computes r_q^{-1} as a common root of $g_i(x), i = \overline{1,t}$ modulo q.

4.1.1 Recovering of modulus p

For computing p in [25] the authors propose to consider the following matrix $\mathbf{A} \in \mathbf{Z}_n^{t \times (d+1)}$:

$$\mathbf{A} = \begin{bmatrix} -a_1 & c_{p,1,1} & \dots & c_{p,1,d} \\ -a_2 & c_{p,2,1} & \dots & c_{p,2,d} \\ \dots & \dots & \dots & \dots \\ -a_t & c_{p,t,1} & \dots & c_{p,t,d} \end{bmatrix}.$$

According to (3) homogeneous system of linear equations $(\mathbf{A} \mid \mathbf{0})$ has a nontrivial solution modulo p:

$$\mathbf{v}^T = (1, r_p^{-1}, (r_p^{-1})^2, \dots, (r_p^{-1})^d).$$

Therefore for $t = d + 1$ $\mathbf{A}$ is a square matrix having zero determinant modulo p. Then equality $\det(\mathbf{A}) = p \cdot s \in \mathbf{Z}_n$, $s \in \{0,1,\dots,q-1\}$ holds. The last one means that if $s \neq 0$ p may be recovered as follows:

$$\mathrm{p} := \mathrm{GCD}(\det(\mathbf{A}), n).$$

According to Chinese reminder theorem we have $\det(\mathbf{A}) = (\det(\mathbf{A}) \bmod q) \cdot p \cdot (p^{-1} \bmod q)$. So $s = 0$ if and only if $\det(\mathbf{A}) \bmod q = 0$. The authors of [25] prove that

$$\Pr(\det(\mathbf{A}) \bmod q \neq 0) > e^{-3/2 \cdot (\mathrm{p}-1)} \quad (5),$$

where for large p value $e^{-3/2\cdot(p-1)} \approx 1$. Thus having $d+1$ pairs (plaintext, ciphertext) A may recover p with probability ≈ 1. Asymptotical complexity of computing p using this method is $O(d^3 \cdot \log^2(n))$.

Remark 3. Inequality (5) in [25] was proven using assumptions that $c_{p,i,j} \xleftarrow{\$} \mathbf{Z}_p$ and $a_i \bmod q \xleftarrow{\$} \mathbf{Z}_q$. But of course this is correct only if probabilistic distribution $\mathbf{D}$ over P is uniform. For not uniform $\mathbf{D}$ (5) is not true. In the worst case $\mathbf{D}$ may be such that $\Pr(0) \approx 1$ and for moderate values of d $\Pr(\det(\mathbf{A}) \bmod q = 0) > 1/2$, because if the first column of $\mathbf{A}$ is a zero vector then $\det(\mathbf{A}) \bmod q = 0$ holds. So for such $\mathbf{D}$ the probability of successful cryptanalysis is not so good. In general additional study is necessary, because it is not immediately clear how to estimate $\Pr(\det(\mathbf{A}) \bmod q \neq 0)$ for arbitrary $\mathbf{D}$.

4.1.2 Recovering of r_p^{-1}, r_q^{-1}

Now we suppose $t = d+1$ and p is recovered using $(a_i \in P, c_i \in C), i = \overline{1,t}$. The first way to compute r_p^{-1} is to solve the system of linear equations $(\mathbf{A} \mid \mathbf{0})$. The second way is to compute:

$$f(x) = GCD(f_{p,1}(x), ..., f_{p,d+1}(x)),$$

where $f_{p,i}(x) \coloneqq f_i(x) \bmod p = c_{p,i}(x) - a_{p.i}$, $a_{p.i} \coloneqq a_i \bmod p$. Obviously

$$f(x) = (x - r_p^{-1}) \cdot GCD(f_{p,1}^0(x), ..., f_{p,d+1}^0(x))$$

holds, where $f_{p,i}^0(x) = f_{p,i}(x) / (x - r_p^{-1}) \in \mathbf{Z}_p[x]$, $i = \overline{1,d+1}$. If $GCD(f_{p,1}^0(x), ..., f_{p,d+1}^0(x)) = 1$ then $f(x) = x - r_p^{-1}$ and therefore r_p^{-1} is recovered. Based on assumption that for all $i = \overline{1,d+1}$: $f_{p,i}^0(x) \xleftarrow{\$} \mathbf{Z}_p[x]$, $\deg(f_{p,i}^0(x)) = d-1$, the authors of [25] give an estimation

$$\Pr(f(x) = x - r_p^{-1}) = \Pr(GCD(f_{p,1}^0(x), ..., f_{p,d+1}^0(x)) = 1) > (1 - 1/p^d)^{d-1}. \quad (6)$$

So for large p and moderate d the probability to recover r_p^{-1} becomes close to 1.

Remark 4. Both ways to compute r_p^{-1} have equivalent complexity $O(d^3 \cdot \log^2(p))$.

In [25] the authors didn't give a proof that all $f_{p,i}^0(x)$ are uniformly random. So here we fill this gap.

Statement 3. Let distribution $\mathbf{D}$ is uniform and let there is a polynomial $f(x) = c_p(x) - a \in \mathbf{Z}_n[x], \deg(f) = d$ constructed using pair $(a, c = (c_p(x), c_q(x))$.

Then $f_p^0(x) = f_p(x)/(x - r_p^{-1}) \in \mathbf{Z}_p[x]$ is uniformly random with $\deg(f_p^0(x)) = d-1$, where $f_p(x) \coloneqq f(x) \bmod p$.

Proof: Let's look at $f_p(x) = \sum_{i=0}^{d} f_{p,i} \cdot x^i \in \mathbf{Z}_p[x]$. According to encryption procedure $f_{p,i} \coloneqq (a'_i \cdot r_p^i) \bmod p, i = \overline{1,d}$ and $f_{p,0} \coloneqq (-\sum_{i=1}^{d} a'_i) \bmod p \ (\coloneqq (-a) \bmod p)$. Using ordinary polynomial division it's easy to verify that $f_p^0(x) = f_p(x)/(x - r_p^{-1}) = \sum_{i=0}^{d-1} f_{p,i}^0 \cdot x^i$, where $f_{p,d-1}^0 \equiv r_p^d \cdot a'_d \pmod p$, $f_{p,d-2}^0 \equiv r_p^{d-1} \cdot (a'_d + a'_{d-1}) \pmod p$, …, $f_{p,1}^0 \equiv r_p^2 \cdot (a'_d + a'_{d-1} + ... + a'_2) \pmod p$ and $f_{p,1}^0 \equiv r_p \cdot (a'_d + a'_{d-1} + ... + a'_1) \pmod p \equiv r_p \cdot a \pmod p$. Coefficients $f_{p,i}^0, i = \overline{0,d-1}$ are independent random values, where $f_{p,i}^0 \xleftarrow{\$} \mathbb{Z}_p, i = \overline{1,d-2}$, $f_{p,d-1}^0 \xleftarrow{\$} \mathbb{Z}_p \setminus \{0\}$, $f_{p,0}^0 \xleftarrow{\mathbf{D}} \mathbb{Z}_p$. So obviously if **D** is uniform then $f_p^0(x) \xleftarrow{\$} \mathbf{Z}_p[x]$ and $\deg(f_p^0(x)) = d-1$. □

One may see that for not uniform **D** polynomials $f_{p,i}^0(x), i = \overline{1,d+1}$ are not uniformly random . And in this case it is not clear whether estimation (6) is true. Thus additional study should be carried out.

Let's turn on to the uniform **D**. We would like to note that in this case instead of estimation (6) one may obtain the exact value of $\Pr(GCD(f_{p,1}^0(x), ..., f_{p,d+1}^0(x)) = 1)$. In [26] the following result based on Euclidean algorithm was proved.

Corollary 1 ([26]). Let $(d_1, ..., d_m)$ be an ordered m-tuple of nonnegative integers (not all zero) and for $1 \le i \le m$ let $a_i(x) \xleftarrow{\$} \mathbf{Z}_p[x]$ $\deg(a_i(x)) = d_i$, where p is a prime. Then the probability that $a_1(x), ..., a_m(x)$ are relatively prime is $1 - 1/p^{m-1}$.

Based on this corollary we have $\Pr(GCD(f_{p,1}^0(x), ..., f_{p,d+1}^0(x)) = 1) = 1 - 1/p^d$ that is ≈ 1 for large p.

Similarly $g(x) = GCD(g_{q,1}(x), ..., g_{q,d+1}(x)) = x - r_q^{-1}$ with probability $1 - 1/q^d$, where $g_i(x) = c_{q,i}(x) - a_i \in \mathbf{Z}_n[x], i = \overline{1,d+1}$, $g_{q,i}(x) \coloneqq g_i(x) \bmod q = c_{q,i}(x) - a_{q,i}$, $a_{q,i} \coloneqq a \bmod q$. And finally we obtain that the probability to recover r_p^{-1}, r_q^{-1} is equal to $(1 - 1/p^d) \cdot (1 - 1/q^d)$. It should be noted that the last one is true because according to encryption procedure for uniform **D** for $\forall i$ polynomials $f_{p,i}(x)$ and $g_{q,i}(x)$ may be considered as independent random polynomials.

Summarizing all said above we see that KPA proposed in [25] requires $t \geq d+1$ *pairs (plaintext, ciphertext) to recover secret key with probability* $\Pr \approx 1$. *But estimation* $\Pr \approx 1$ *is proved only for uniform* $\mathbf{D}$. *The total asymptotical complexity of KPA is* $O(d^3 \cdot \log^2(n))$.

4.2 Our improvement of KPA

Now we discuss how to reduce the number of pairs t necessary for successful KPA on cryptosystem [10]. First we recall the notion of resultant for two polynomials. Let there are $f(x) = \sum_{i=0}^{d_1} f_i \cdot x^i, g(x) = \sum_{i=0}^{d_2} g_i \cdot x^i \in \mathbf{Z}_n[x]$. One may compose a Sylvester matrix $\mathbf{S} \in \mathbf{Z}_n^{(d_1+d_2)\times(d_1+d_2)}$ for $f(x), g(x)$:

$$\mathbf{S} = \begin{vmatrix} f_0 & \dots & f_{d_1} & 0 & 0 & \dots & 0 \\ 0 & f_0 & \dots & f_{d_1} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 0 & f_0 & \dots & f_{d_1} \\ g_0 & \dots & g_{d_2} & 0 & 0 & \dots & 0 \\ 0 & g_0 & \dots & g_{d_2} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 0 & g_0 & \dots & g_{d_2} \end{vmatrix}. \quad (7)$$

The resultant of polynomials $f(x), g(x) \in \mathbf{Z}_n[x]$ is defined as follows: $\Theta = \mathrm{Res}(f(x), g(x)) = \det(\mathbf{S}) \bmod n \in \mathbf{Z}_n$. It is well known result that $\Theta = 0$ if and only if $f(x)$ and $g(x)$ have at least one common root or factor modulo n (for details see [27]). For further discussion we need the following simple statements.

Statement 4 . If for $n = p \cdot q$ polynomials $f(x), g(x) \in \mathbf{Z}_n[x]$ have at least one common root or factor modulo p (or q) then $\Theta_p = 0$ (or $\Theta_q = 0$), where $\Theta_p := \Theta \bmod p$, $\Theta_q := \Theta \bmod q$.

Statement 5. If $n = p \cdot q$, where $p \neq q, GCD(p, q) = 1$, then $\Theta = 0$ if and only if $\Theta_p = 0$, $\Theta_q = 0$.

We skip the proof because this statements may be immediately derived from Chinese reminder theorem and congruences properties.

Let's return to KPA on cryptosystem [10]. Now we will demonstrate that interception only of two pairs (plaintext, ciphertext) may be enough to recover factorization of n and $k = (r_p, r_q)$.

4.2.1 Recovering of modulus p

Let's suppose A intercepted $(a_i, c_i = (c_{p,i}(x) \in \mathbf{Z}_p[x],\ c_{q,i}(x) \in \mathbf{Z}_q[x])), i = \overline{1,2}$, where $\deg(c_{p,i}(x)) = d, \deg(c_{q,i}(x)) = d$. Let's look at the resultant $\Theta = \mathrm{Re}s(f_1(x), f_2(x)) \in \mathbf{Z}_n$, where $f_i(x) = c_{p,i}(x) - a_i \in \mathbf{Z}_n[x], i = 1,2$. As we've already seen $f_1(x), f_2(x)$ have a common root r_p^{-1} modulo p. According to statement 4 $\Theta_p = 0$ and hence $\Theta = p \cdot s, s \in \{0,1,...,q-1\}$. So for $s \neq 0$ A can compute p according formula:

$$\mathrm{p} := \mathrm{GCD}(\Theta, n).$$

Please note that the last one is true because here q is prime and $\mathrm{GCD}(s,q) = 1$ for $s \neq 0, q$.

As a result we obtain that to recover p it's enough to have only two pairs $(a_i, c_i), i = \overline{1,2}$ with $\Theta \neq 0$. So it's necessary to find out how much the probability $\mathrm{Pr}_0 = \Pr(\Theta \neq 0)$ for randomly intercepted pairs. To estimate Pr_0 we should note that according to statement 5 $\Theta = 0$ if and only if $\Theta_q = 0$ and then $\mathrm{Pr}_0 = \Pr(\Theta_q \neq 0)$. Obviously $\Theta_q \neq 0$ if and only if $GCD(f_{q,1}(x), f_{q,2}(x)) = 1$, where $f_{q,i}(x) = f_i(x) \bmod q \in \square_q[x], i = 1,2$. If $f_{q,1}(x), f_{q,2}(x)$ were uniformly random in $\square_q[x]$ then $\mathrm{Pr}_0 = \Pr(\Theta_q \neq 0)$ would be equal to $1 - 1/q$ according to corollary 1. But unfortunately in fact $f_{q,\mathrm{i}}(x) = \sum_{j=0}^{d} f_{q,i,j} \cdot x^j, i = 1,2$ are not strictly uniform even if distribution **D** is uniform. Indeed for uniform **D** there are $f_{q,i,j} \xleftarrow{\$} \{0,1,..., p-1\}, j = \overline{1, d-1}$, $f_{q,i,\mathrm{d}} \xleftarrow{\$} \{1,..., p-1\}$ and $f_{q,i,0} \xleftarrow{\$} \mathbf{Z}_q$. Estimation

$$\mathrm{Pr}_0 \approx 1 - 1/q \quad (8)$$

we are not ready to prove now. But (8) correlates very good with computer experiments. In tables 1,2 we present practical estimation of Pr_0 for uniform **D** for different d.

Remark 5. Cryptosystem from [10] and presented KPA were implemented using Qt 1.3.1 and NTL library [28]. For practical estimation of Pr_0 two pairs (a_i, c_i) were generated randomly 10^5 times. Then the number of cases with $\Theta_q \neq 0$ was counted. The case of not uniform **D** should be studied additionally. The only thing we can say now that in the worst case **D** may be such that $\Pr(0) = \beta$, where $\beta \approx 1$ and then $\mathrm{Pr}_0 = \Pr(\Theta_q = 0) > \beta^2$ that is ≈ 1. So for such **D** this KPA fails with overwhelming probability.

Table 1. Estimations of $\Pr_0$ *for different* p,q *and* $d=10$.

n	p	q	Practical estimation of $\Pr_0$	$1-1/q$
6	2	3	0.67	0.67
35	5	7	0.86	0.86
91	7	13	0.922	0.923
253	11	23	0.956	0.957
1517	37	41	0.97	0.97
3599	59	61	0.98	0.99
9991	97	103	0.99	0.991

Table 2. Estimations of $\Pr_0$ *for different* p,q *and* $d=50$.

n	p	q	Practical estimation of $\Pr_0$	$1-1/q$
15	3	5	0.8	0.8
221	13	17	0.92	0.94
1147	31	37	0.954	0.972
2173	41	53	0.999	0.999
13943	103	131	0.999	0.999

The asymptotical complexity of this method to recover p is $O(d^3 \cdot \log^2(n))$.

Finally we would like to note that the idea to compute resultant of polynomials for recovering p we borrow from [29]. In [29] the author presented KPA on another Doming-Ferrer homomorphic cryptosystem [11]. Encryption in [11] works similar to [10]. Plaintext $a \in \mathbf{Z}_{n'}$ first is mapped into random polynomial $a'(x) \in \mathbf{Z}_{n'}[x]$ such that $a'(1) \equiv a (\mathrm{mod}\, n')$, $\deg(a'(x)) = d, a'(0) = 0$. Ciphertext is a polynomial $c(x) \in \mathbf{Z}_n[x]$ such that $c(x) \coloneqq a'(r \cdot x) \bmod n$, where $r \in \mathbf{Z}_n^*$ – secret key, n – big integer ($\log(n) \approx 1000$) with many small divisors, $n' \mid n$ and $\log(n') \approx 100$. Modulus n' is hidden and n is public. It should be pointed out that in spite of similarity construction from [10] is not a special case of [11] and vice versa.

To break cryptosystem [11] A first should compute n' and second $(r')^{-1} \coloneqq r^{-1} \bmod n'$ as a common root of polynomials $f_i(x) = c_i(x) - a_i \in \mathbf{Z}_n[x], i = \overline{1,t}$ modulo n'. According to congruences properties $(r')^{-1}$ may be used for decryption instead of r^{-1}. For recovering n' in [29] the author proposes to compute $n'' = GCD(n, \mathrm{Res}(f_1, f_2), \mathrm{Res}(f_3, f_3), ..., \mathrm{Res}(f_{t-1}, f_t))$. Obviously

$\Pr(n'' = n') = \Pr(GCD(n / n', \mathrm{Res}(f_1, f_2) / n', \mathrm{Res}(f_3, f_3) / n', ..., \mathrm{Res}(f_{t-1}, f_t) / n') = 1)$

(/ is integer division) holds. Here in contrast to [10] it's not enough to take $t = 2$,

because n has many small divisors. So to estimate $\Pr_0 = \Pr(GCD(n/n', \mathrm{Re}s(f_1, f_2)/\mathrm{n}', \mathrm{Re}s(f_3, f_3)/\mathrm{n}', ..., \mathrm{Re}s(f_{t-1}, f_t)/\mathrm{n}') = 1)$ one should involve a known result about the probability that randomly chosen integers are coprime. According to this result $\Pr_0 \approx 1/\zeta(\mathrm{t}/2+1)$ holds (we suppose t is even), where ζ is Riemann's zeta function. So for $t = 2$ we have $\Pr_0 \approx 0,61$. That is not enough of course. To obtain $\Pr_0 \approx 1$ one should take $t > 100$.

Summarizing all said above we would like to stress out that idea of computing resultants doesn't work so good for cryptosystem [11], because A *must intercept many pairs to recover secret modulus with overwhelming probability. But for [10] computing resultant allows to decrease* t *meaningfully. Now the only case in which we while don't know how to find* p *is* $t = 1$.

4.2.2 Recovering of r_p^{-1}, r_q^{-1}

For recovering r_p^{-1} A may compute

$$f(x) = GCD(f_{p,1}(x), f_{p,2}(x)) \in \mathbf{Z}_p[x],$$

where $f_{p,i}(x) := f_i(x) \bmod p$, $f_i(x) = c_{p,i}(x) - a_i \in \mathbf{Z}_n[x], i = 1,2$. For uniform **D** according to corollary 1 we obtain $\Pr(f(x) = x - r_p^{-1}) = 1 - 1/p$ that is ≈ 1 for large p. Similarly r_q^{-1} may recovered with probability $1 - 1/\mathrm{q}$. So the total probability to find r_p^{-1}, r_q^{-1} now is $\Pr_1 = (1 - 1/p) \cdot (1 - 1/\mathrm{q})$. The last one is ≈ 1 for large p, q.

The asymptotical complexity of computing r_p^{-1}, r_q^{-1} now is $O(d^2 \cdot \log^2(q))$.

To conclude we would like to present the total running time T of our KPA (time to recover p, q and r_p^{-1}, r_q^{-1}). Time measurements were done using PC with the following characteristics: Quad Core Celerone 1,7 GHz with 4 GB memory.

Table 3. Running time of KPA.

d	T for $\log n = 2^{10}, \log p = 2^9$	T for $\log n = 2^{11}, \log p = 2^{10}$
8	38 ms	112 ms
16	121 ms	387 ms
32	460 ms	1.5 s
64	1.9 s	6 s
128	9.5 s	27 s
256	52 s	2 min
512	5 min	12 min
1024	22 min	50 min

5. Conclusion

We have analysed the existing method [25] of known plaintext cryptanalysis of Domingo-Ferrer homomorphic cryptosystem [10]. This analysis shows that it provably works with overwhelming probability only for uniform probabilistic distribution $\mathbf{D}$ over plaintexts space. The case of arbitrary $\mathbf{D}$ requires the further study. Also based on results obtained in [29] we slightly modified KPA from [25]. The obtained KPA works successful even for the number t of intercepted pairs (plaintext, ciphertext) equal to 2. This is in contrast to [25] where $t \geq d+1$ must be satisfied. But unfortunately our attack also provably recovers secret parameters with probability ≈ 1 only for uniform $\mathbf{D}$. And the case of arbitrary $\mathbf{D}$ also should be studied additionally. If $\mathbf{D}$ is such that $\Pr(0) \approx 1$ than both attack fails with probability close to 1. In future we are planning to investigate the resistance of Domingo-Ferrer homomorphic cryptosystem to ciphertext only attack.

References

[1]. R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 1978, vol. 21, no. 2, pp. 120–126.

[2]. S. Goldwasser and S. Micali. Probabilistic encryption & how to play mental poker keeping secret all partial information. Proceedings of the fourteenth annual ACM symposium on Theory of computing. ACM, 1982, pp. 365–377.

[3]. P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. Advances in cryptologyEUROCRYPT99. Springer, 1999, pp. 223–238.

[4]. D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-dnf formulas on ciphertexts. Theory of cryptography. Springer, 2005, pp. 325–341.

[5]. Damgård I., Jurik M. A generalisation, a simplification and some applications of Paillier's probabilistic public-key system . Public Key Cryptography. – Springer Berlin Heidelberg, 2001, pp. 119-136.

[6]. Rivest R. L., Adleman L., Dertouzos M. L. On data banks and privacy homomorphisms . Foundations of secure computation, 1978, vol. 4, no. 11, pp. 169-180.

[7]. Brickell E. F., Yacobi Y. On privacy homomorphisms . Advances in Cryptology—EUROCRYPT'87. – Springer Berlin Heidelberg, 1988, pp. 117-125.

[8]. Fellows M., Koblitz N. Combinatorial cryptosystems galore //Contemporary Mathematics, 1993, vol. 168, no. 2, pp. 51-61.

[9]. O. Zhirov, O. V. Zhirova, and S. F. Krendelev. Bezopasnye oblachnye vychisleniya s pomoshh'yu gomomorfnoj kriptografii. [Secure cloud computing using homomorphic cryptography]. Bezopasnost' informatsionnykh tekhnologij. [The security of information technologies], vol. 1, pp. 6–12, 2013 (in Russian).

[10]. J. D. i. Ferrer, A new privacy homomorphism and applications. Information Processing Letters, vol. 60, no. 5, pp. 277–282, 1996.

[11]. J. Domingo-Ferrer. A provably secure additive and multiplicative privacy homomorphism. Information Security. Springer, 2002, pp.471–483.

[12]. A. Trepacheva and L. Babenko. Known plaintexts attack on polynomial based homomorphic encryption. Proceedings of the Seventh International Conference on Security of Information and Networks. ACM, 2014.

[13]. M. R. Albrecht, P. Farshim, J.-C. Faugere, and L. Perret. Polly cracker, revisited. Advances in Cryptology–ASIACRYPT 2011. Springer, 2011, pp. 179–196.
[14]. C. Gentry. A fully homomorphic encryption scheme. Ph.D. dissertation, Stanford University, 2009.
[15]. M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. in Advances in Cryptology–EUROCRYPT 2010. Springer, 2010, pp. 24–43.
[16]. Z. Brakerski, C. Gentry, and V. Vaikuntanathan,. (Leveled) Fully homomorphic encryption without bootstrapping. Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. ACM, 2012, pp. 309–325.
[17]. N. P. Smart and F. Vercauteren. Fully homomorphic SIMD operations. Designs, Codes and Cryptography, pp. 1–25, 2011.
[18]. M. Naehrig, K. Lauter, and V. Vaikuntanathan. Can homomorphic encryption be practical?. Proceedings of the 3rd ACM workshop on Cloud computing security workshop. ACM, 2011, pp. 113–124.
[19]. L. Ertaul and J. H. Yang. Implementation of domingo ferrer's a new privacy homomorphism (df a new ph) in securing wireless sensor networks (wsn)/ Security and Management. Citeseer, 2008, pp. 498–504.
[20]. L. Ertaul, Vaidehi. Implementation of Homomorphic Encryption Schemes for Secure Packet Forwarding in Mobile Ad Hoc Networks (MANETs). IJCSNS International Journal of Computer Science and Network Security, 2007, vol. 7, no. 11 pp. 132-141.
[21]. V. Jariwala and D. Jinwala. Evaluating homomorphic encryption algorithms for privacy in wireless sensor networks. International Journal of Advancements in Computing Technology, vol. 3, no. 6, 2011.
[22]. Vaghasia and K. Bathwar.Public key encryption algorithms for wireless sensor networks in tinyos. IJITEE, 2013, vol. 2, no. 4.
[23]. Sorniotti, L. Gomez, K. Wrona, and L. Odorico. Secure and trusted in-network data processing in wireless sensor networks: a survey. Journal of Information Assurance and Security, 2007, vol. 2, no. 3, pp. 189–199.
[24]. Westhoff, J. Girao, and M. Acharya. Concealed data aggregation for reverse multicast traffic in sensor networks: Encryption, key distribution, and routing adaptation. Mobile Computing, IEEE Transactions on , 2006, vol. 5, no. 10, pp. 1417–1431.
[25]. J. H. Cheon, W.-H. Kim, and H. S. Nam. Known-plaintext cryptanalysis of the domingo-ferrer algebraic privacy homomorphism scheme. Information Processing Letters, 2006, vol. 97, no. 3, pp. 118–123.
[26]. T. Benjamin and C. D. Bennett. The probability of relatively prime polynomials. Mathematics Magazine, 2007, pp. 196–202.
[27]. Davenport, James H., Y. Siret, and E. Tournier. *Computer algebra*. London: Academic Press, 1988, 263 p.
[28]. Shoup V. NTL: A library for doing number theory. – 2001.
[29]. Wagner. Cryptanalysis of an algebraic privacy homomorphism. Information Security. Springer, 2003, pp. 234–239.

Улучшенная атака по известным открытым текстам на гомоморфную криптосистему Доминго-Феррера

А.В. Трепачева <alina1989malina@ya.ru>
Южный федеральный университет,
Россия, 344006, г. Ростов-на-Дону, ул. Большая Садовая 105/42.

Аннотация. Данная работа посвящена криптоанализу по известным открытым текстам гомоморфной криптосистемы, предложенной Доминго-Феррером. В предыдущих работах было показано, что для раскрытия секретного ключа необходимо перехватить по меньшей мере $d+1$ пару (открытый текст, шифртекст), где d – степень полиномов, являющихся шифртекстами. Здесь мы проводим анализ существующей атаки по известным открытым текстам, а также показываем, как можно её модифицировать так, чтобы значительно уменьшить нужное количество перехваченных пар. А именно, оказывается, что достаточно всего лишь двух пар для раскрытия секретного ключа. Время работы предложенной атаки так же, как и для уже существующей, зависит полиномиально от d и логарифмически от размера пространства открытых текстов. Представлены результаты компьютерных экспериментов.

Ключевые слова: атака по известным открытым текстам; гомоморфное шифрование; облачные вычисления.

Литература

[1]. R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 1978, vol. 21, no. 2, pp. 120–126.
[2]. S. Goldwasser and S. Micali. Probabilistic encryption & how to play mental poker keeping secret all partial information. Proceedings of the fourteenth annual ACM symposium on Theory of computing. ACM, 1982, pp. 365–377.
[3]. P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. Advances in cryptologyEUROCRYPT99. Springer, 1999, pp. 223–238.
[4]. D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-dnf formulas on ciphertexts. Theory of cryptography. Springer, 2005, pp. 325–341.
[5]. Damgård I., Jurik M. A generalisation, a simplification and some applications of Paillier's probabilistic public-key system . Public Key Cryptography. – Springer Berlin Heidelberg, 2001, pp. 119-136.

[6]. Rivest R. L., Adleman L., Dertouzos M. L. On data banks and privacy homomorphisms . Foundations of secure computation, 1978, vol. 4, no. 11, pp. 169-180.
[7]. Brickell E. F., Yacobi Y. On privacy homomorphisms . Advances in Cryptology—EUROCRYPT'87. – Springer Berlin Heidelberg, 1988, pp. 117-125.
[8]. Fellows M., Koblitz N. Combinatorial cryptosystems galore //Contemporary Mathematics, 1993, vol. 168, no. 2, pp. 51-61.
[9]. Жиров А.О., Жирова О.В., Кренделев С.Ф.. Безопасные облачные вычисления с помощью гомоморфной криптографии. Безопасность информационных технологий, 2013, Т. 1, С. 6–12.
[10]. J. D. i. Ferrer, A new privacy homomorphism and applications. Information Processing Letters, vol. 60, no. 5, pp. 277–282, 1996.
[11]. J. Domingo-Ferrer. A provably secure additive and multiplicative privacy homomorphism. Information Security. Springer, 2002, pp.471–483.
[12]. A. Trepacheva and L. Babenko. Known plaintexts attack on polynomial based homomorphic encryption. Proceedings of the Seventh International Conference on Security of Information and Networks. ACM, 2014.
[13]. M. R. Albrecht, P. Farshim, J.-C. Faugere, and L. Perret. Polly cracker, revisited. Advances in Cryptology–ASIACRYPT 2011. Springer, 2011, pp. 179–196.
[14]. C. Gentry. A fully homomorphic encryption scheme. Ph.D. dissertation, Stanford University, 2009.
[15]. M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. in Advances in Cryptology–EUROCRYPT 2010. Springer, 2010, pp. 24–43.
[16]. Z. Brakerski, C. Gentry, and V. Vaikuntanathan,. (Leveled) Fully homomorphic encryption without bootstrapping. Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. ACM, 2012, pp. 309–325.
[17]. N. P. Smart and F. Vercauteren. Fully homomorphic SIMD operations. Designs, Codes and Cryptography, pp. 1–25, 2011.
[18]. M. Naehrig, K. Lauter, and V. Vaikuntanathan. Can homomorphic encryption be practical?. Proceedings of the 3rd ACM workshop on Cloud computing security workshop. ACM, 2011, pp. 113–124.
[19]. L. Ertaul and J. H. Yang. Implementation of domingo ferrer's a new privacy homomorphism (df a new ph) in securing wireless sensor networks (wsn)/ Security and Management. Citeseer, 2008, pp. 498–504.
[20]. L. Ertaul, Vaidehi. Implementation of Homomorphic Encryption Schemes for Secure Packet Forwarding in Mobile Ad Hoc Networks (MANETs). IJCSNS International Journal of Computer Science and Network Security, 2007, vol. 7, no. 11 pp. 132-141.
[21]. V. Jariwala and D. Jinwala. Evaluating homomorphic encryption algorithms for privacy in wireless sensor networks. International Journal of Advancements in Computing Technology, vol. 3, no. 6, 2011.
[22]. Vaghasia and K. Bathwar.Public key encryption algorithms for wireless sensor networks in tinyos. IJITEE, 2013, vol. 2, no. 4.
[23]. Sorniotti, L. Gomez, K. Wrona, and L. Odorico. Secure and trusted in-network data processing in wireless sensor networks: a survey. Journal of Information Assurance and Security, 2007, vol. 2, no. 3, pp. 189–199.
[24]. Westhoff, J. Girao, and M. Acharya. Concealed data aggregation for reverse multicast traffic in sensor networks: Encryption, key distribution, and routing adaptation. Mobile Computing, IEEE Transactions on , 2006, vol. 5, no. 10, pp. 1417–1431.

[25]. J. H. Cheon, W.-H. Kim, and H. S. Nam. Known-plaintext cryptanalysis of the domingo-ferrer algebraic privacy homomorphism scheme. Information Processing Letters, 2006, vol. 97, no. 3, pp. 118–123.
[26]. T. Benjamin and C. D. Bennett. The probability of relatively prime polynomials. Mathematics Magazine, 2007, pp. 196–202
[27]. Davenport, James H., Y. Siret, and E. Tournier. *Computer algebra*. London: Academic Press, 1988, 263 p.
[28]. Shoup V. NTL: A library for doing number theory. – 2001.
[29]. Wagner. Cryptanalysis of an algebraic privacy homomorphism. Information Security. Springer, 2003, pp. 234–239.

Пакетное симметричное полностью гомоморфное шифрование на основе матричных полиномов[1]

Ф.Б. Буртыка <bbfilipp@ya.ru>
Южный федеральный университет,
344006, Россия, г. Ростов-на-Дону, ул. Большая Садовая, д. 105/42

Аннотация. Методы полностью гомоморфного шифрования (ПГШ) – общепризнанный способ организации криптографической защиты облачных вычислений. Однако существующие криптосхемы ПГШ по своим характеристикам не достаточны для применения на практике – одни криптосхемы имеют слишком малую криптостойкость, другие требуют слишком больших вычислительных ресурсов. Для развития последних исследователями из IBM был предложен метод «упаковывания шифртекстов», который был применен ими к криптосхеме с открытым ключом, стойкость которой основана на сложности задач теории решеток. В данной работе метод «упаковки шифртекстов» применен к симметричной криптосхеме на основе матричных полиномов: приводится описание возможных способов организации такой упаковки, представлено описание одного из вариантов таких криптосистем с оценкой сложности алгоритма умножения шифртекстов. В заключение приведено сравнение эффективности полученной криптосхемы с криптосхемами исследователей из IBM.

Ключевые слова: защита информации; облачные вычисления; полностью гомоморфное шифрование; «упаковка шифртекстов»; матричные полиномы; вычисления над зашифрованными данными.

1. Введение

В связи с необходимостью борьбы с угрозами безопасности для облачных вычислений актуальна задача построения эффективных и криптостойких методов полностью гомоморфного шифрования (ПГШ) [1-5]. Такое шифрование позволяет производить любые операции с зашифрованными данными и получать зашифрованный результат, который соответствует результату операций, выполняемых с открытыми данными. Задача построения ПГШ впервые была поставлена в работе [1], но принципиально решена лишь в работе Крейга Джентри [2], где было описано построение алгоритмов ПГШ,

[1] Работа выполнена при поддержке гранта РФФИ №15-07-00597 А «Разработка и исследование алгоритмов полностью гомоморфного шифрования»

сложность которых была полиномиальна от размеров входных данных, а задача взлома сводилась к сложным задачам теории решеток. Эта конструкция, однако, имела лишь теоретическое значение из-за низкой *практической* эффективности алгоритмов ПГШ. Вскоре после [2] последовала серия работ, направленных на улучшение исходных алгоритмов ПГШ Джентри [6-13]. Однако ни в одной из этих работ не было предложено решения пригодного для практического использования. Среди альтернатив криптосхемам Джентри можно упомянуть криптосистему с открытым ключом Polly Cracker [18,19] Нила Коблица и симметричные криптосистемы Доминго-Феррера [16], Ростовцева [21], Кренделева [5,14], Пуульпановой и Хойсика [15]. Однако все эти криптосхемы либо еще менее эффективны, чем схема Джентри, либо имеют невысокую криптостойкость [22, 23]. В [24] и [25] были предложены криптосхемы ПГШ на основе матричных полиномов, которые отличаются высокой эффективностью при предположительно значительной криптостойкости.

В данной работе предлагается повысить эффективность ПГШ на основе матричных полиномов [24,25] с помощью метода упаковки в один шифртекст нескольких открытых текстов с последующей «пакетной» обработкой зашифрованных данных. Данный метод впервые был введен в работах Джентри для ускорения работы его конструкций. И несмотря на то, что даже с использованием этого метода криптосистемы типа Джентри не стали пригодными для практики, их эффективность за счет него на порядок увеличилась. Метод «упаковки шифртекстов» является очень перспективным. Поскольку пакетная обработка подразумевает, что при одной операции над двумя шифртекстами происходит одновременное выполнение операций по-координатно над всеми содержащимися в этих шифртекстах открытыми текстами (SIMD организация обработки).

Статья организована следующим образом. В разделе 2 приведены необходимые обозначения, теоретические сведения и определение гомоморфного шифрования. В разделе 3 более подробно описывается метод «упаковки шифртекстов» В разделе 4 показаны различные способы модификации ПГШ на основе матричных полиномов в ПГШ с возможностью «упаковки шифртекстов». В разделе 5 приведены экспериментальные данные по реализации описанных криптосхем на матричных полиномах, а также сравнение по производительности с криптосхемами Джентри, Бракерски и Вэйкунтанасана из работ [8,26].

2. Основные определения и обозначения

Далее в статье множество натуральных чисел будем обозначать как $\mathbf{N}$, кольцо классов вычетов по модулю p будем обозначать как $\mathbf{Z}_p$ (в статье будем использовать только кольца по простому модулю для облегчения доказательств). Прописными греческими буквами (например,) будут

обозначаться различные параметры, при этом λ будет всегда обозначать параметр уровня криптостойкости. Матрицы будут обозначаться заглавными латинскими буквами полужирным шрифтом (например, $\mathbf{A},\mathbf{B}$), при этом единичную матрицу будем обозначать как $\mathbf{I}$. Векторы будут обозначаться прописными латинскими буквами со стрелкой над ними, например . Обозначим через $\mathbf{Z}_p^{N\times N}$ кольцо $N\times N$ матриц с элементами из кольца $\mathbf{Z}_p$. Напомним, что *спектром* матрицы $\mathbf{A}$ (обозначение $Sp(\mathbf{A})$) называется множество собственных векторов матрицы $\mathbf{A}$, т.е. таких векторов $\vec{v}$ для которых $\mathbf{A}\cdot\vec{v}=a\cdot\vec{v}$ при некотором $a\in\mathbf{Z}_p$. Множество матриц коммутирующих с матрицей $\mathbf{A}$ будем называть коммутантом матрицы и обозначать $Comm(\mathbf{A})$. При описании алгоритмов будем использовать запись $x\xleftarrow{\$}X$ для обозначения того, что *x* выбрано случайно по равномерному распределению из конечного множества *X*. Также в статье будет использовано понятие схемы из функциональных элементов (СФЭ), для которой нам будет достаточно знать что это вектор-функция над векторами фиксированной размерности с элементами из $\mathbf{Z}_p$.

2.1 Матричные полиномы

Рассмотрим множество последовательностей матриц из $\mathbf{Z}_p^{N\times N}$:

$$F=\{\mathbf{A}_0,\mathbf{A}_1,\mathbf{A}_2,\ldots\},\mathbf{A}_i\in\mathbf{Z}_p^{N\times N},$$

таких, что все $\mathbf{A}_i$, кроме конечного их числа, равны нулевой матрице. Пусть $\mathbf{Z}_p^{N\times N}[X]$ обозначает множество всех таких последовательностей. Если $F,G\in\mathbf{Z}_p^{N\times N}[X]$, $G=\{\mathbf{B}_0,\mathbf{B}_1,\mathbf{B}_2,\ldots\},\mathbf{B}_i\in\mathbf{Z}_p^{N\times N}$, то определим

$$\begin{aligned}&F+G=\{\mathbf{A}_0+\mathbf{B}_0,\mathbf{A}_1+\mathbf{B}_1,\mathbf{A}_2+\mathbf{B}_2,\ldots\},\\&F\cdot G=\{\mathbf{A}_0\cdot\mathbf{B}_0,\mathbf{A}_0\cdot\mathbf{B}_1+\mathbf{A}_1\cdot\mathbf{B}_0,\mathbf{A}_0\cdot\mathbf{B}_2+\mathbf{A}_1\cdot\mathbf{B}_1+\mathbf{A}_2\cdot\mathbf{B}_0,\ldots\}=\{\mathbf{C}_k\},\\&\text{где }\mathbf{C}_k=\sum_{i+j=k}\mathbf{A}_i\cdot\mathbf{B}_j,k=0,1,2,\ldots.\end{aligned}\qquad(1)$$

Можно показать, что при таких определениях сложения и умножения множество $\mathbf{Z}_p^{N\times N}[X]$ становится кольцом. Элементы этого кольца будем называть *матричными полиномами.*

Лемма 1. *Матричные полиномы образуют (ассоциативное) кольцо.*
Доказательство. Выполняется непосредственной проверкой аксиом кольца.

Приведенный матричный полином – это такой полином, у которого коэффициент при старшей степени равен единичной матрице. Также для дальнейшего важным элементом является *деление матричных полиномов*.

Теорема 1 (О делении матричных полиномов, [29,30]) Пусть $M(X) = X^m + \mathbf{A}_{m-1} \cdot X^{m-1} + ... + \mathbf{A}_0$ и $W(X) = X^p + \mathbf{B}_{p-1} \cdot X^{p-1} + ... + \mathbf{B}_0$, при $m \geq p$. Тогда существуют единственный приведенный матричный полином *F(X)* степени $m–p$ и единственный матричный полином *L(X)* степени $p–1$ такие что

$$M(X) = F(X) \cdot X + \mathbf{B}_p \cdot F(X) \cdot X^p + ... + \mathbf{B}_1 \cdot F(X) + L(X). \quad (2)$$

Доказательство. Пусть $F(X) = X^{m-p} + \mathbf{F}_{m-p-1} \cdot X^{m-p-1} + ... + \mathbf{F}_0$, и $L(X) = X^p + \mathbf{L}_{p-1} \cdot X^{p-1} + ... + \mathbf{L}_0$. Приравнивая коэффициенты в равенстве (2), $\mathbf{F}_0, \mathbf{F}_1, ..., \mathbf{F}_{m-p-1}$ и $\mathbf{L}_0, \mathbf{L}_1, ..., \mathbf{L}_{p-1}$ могут быть определены из полученной системы m матричных уравнений.

Следствие: каждый приведенный матричный полином порождает (левосторонний) идеал в кольце матричных полиномов.

Каждому матричному полиному $\mathbf{P}(X)$ можно естественным образом сопоставить матричное уравнение $\mathbf{P}(X) = \mathbf{0}$. Интересно, что такое матричное уравнение может иметь корней больше, чем его степень [29,30], а может и не иметь корней совсем. В случае если такое уравнение не имеет корней, соответствующий матричный полином будем называть *неприводимым*. Множество корней матричного полинома $\mathbf{P}(X)$ будем обозначать через $roots\left(\mathbf{P}(X)\right)$. Матричный полином, являющийся одновременно неприводимым и приведенным будем называть *примитивным*.

2.2 Определения гомоморфного шифрования

Общая организация системы защищенных вычислений с помощью симметричного гомоморфного шифрования будет идентична описанной в [25]. Для удобства дальнейшего изложения введем некоторые формальные определения, связанные с такой системой шифрования: гомоморфная криптосхема ε представляет собой четвёрку алгоритмов $\left(\mathrm{KeyGen}_\varepsilon, \mathrm{Encrypt}_\varepsilon, \mathrm{Decrypt}_\varepsilon, \mathrm{Evaluate}_\varepsilon\right)$. Вероятностный алгоритм $\mathrm{KeyGen}_\varepsilon$ принимает на вход параметр уровня криптостойкости λ и выдает в качестве результата пару ключей **(sk, rk)**, где **sk** – секретный ключ, который хранится у клиента, а **rk** – ключ перешифрования, передаваемый серверу (он позволяет серверу сокращать размер шифртекстов в процессе вычислений, но

не позволяет зашифровывать или расшифровывать). Алгоритмы $\mathrm{Encrypt}_\varepsilon$ и $\mathrm{Decrypt}_\varepsilon$ принимают на вход, соответственно, шифртекст или открытый текст вместе с секретным ключом **sk**. Алгоритм $\mathrm{Evaluate}_\varepsilon$ принимает на вход СФЭ F, набор шифртекстов $\langle m_1,...,m_t \rangle$, ключ перешифрования **rk**, и выдает в качестве результата шифртекст *c*. Вычислительная сложность всех этих алгоритмов должна быть полиномиальна от параметра уровня криптостойкости λ и (в случае алгоритма $\mathrm{Evaluate}_\varepsilon$) количества схемных элементов F, а также они должны удовлетворять приведенным ниже требованиям корректности.

Определение 1. (Корректность расшифрования после гомоморфного вычисления). Криптосхема $\varepsilon = (\mathrm{KeyGen}, \mathrm{Encrypt}, \mathrm{Decrypt}, \mathrm{Evaluate})$ корректна для СФЭ F, имеющей t входов, если для любой пары ключей **(sk, rk)**, выданной алгоритмом $\mathrm{KeyGen}(\lambda)$, любых t открытых текстов m_i и соответствующих им шифртекстов $c_i \leftarrow \mathrm{Encrypt}(\mathbf{sk}, m_i)$ выполняется:

$$\mathrm{Decrypt}\left(\mathbf{sk}, \mathrm{Evaluate}(\mathbf{rk}, F, c)\right) = F(m_1,...,m_t).$$

Определение 2. Криптосхема $\varepsilon = (\mathrm{KeyGen}, \mathrm{Encrypt}, \mathrm{Decrypt}, \mathrm{Evaluate})$ полностью гомоморфна для класса СФЭ, если она корректна для всех СФЭ из этого класса.

Определение 3. Гомоморфная криптосхема называется *компактной*, если размер шифртекстов, получающихся в результате гомоморфного вычисления произвольной функции f над шифртекстами, не зависит от размера схемы из функциональных элементов, представляющей f, и ограничен полиномом $\beta(\lambda)$.

Замечание: вышеприведенному определению компактности не удовлетворяет, например, криптосистема из [15] или криптосистема с булевыми полиномами [5] поскольку размер шифртекстов в них хотя в общем и ограничен, но это ограничение экпоненциально (не полиномиально).

Системы определений альтернативные вышеприведенной можно найти в [15] и [14], где оно было введено через понятия *открытого и секретного идеалов в кольце*.

3. Гомоморфное шифрование и метод упаковки шифртекстов

Впервые идея проведения векторных (SIMD) операций над зашифрованными данными была высказана в работе Смарта и Веркотерена [6]. В [6] было замечено, что с применением китайской теоремы об остатках, пространство открытых текстов некоторых известных к тому времени криптосхем ПГШ может быть расширено за счет введения векторов, компоненты которых – «ячейки» для отдельных открытых текстов (plaintext slots). При этом одно гомоморфное сложение (Add) или умножение (Mult) пары шифртекстов неявно складывает или умножает (по-компонентно) векторы открытых текстов целиком.

Каждая ячейка для открытого текста предназначается для хранения элемента из какого-то конечного поля $\mathbf{K}_n = \mathrm{F}_{p^n}$, и, абстрактно, если есть два шифртекста, которые хранят (зашифрованные) сообщения $m_0,...,m_{l-1} \in \mathbf{K}_n^l$ и $m'_0,...,m'_{l-1} \in \mathbf{K}_n^l$ соответственно в ячейках $0,...,l-1$ открытого текста, в результате применения *l*-арного сложения к двум шифртекстам получается новый шифртекст, хранящий $m_0 + m'_0,...,m_{l-1} + m'_{l-1} \in \mathbf{K}_n^l$, а применение *l*-арного умножения двух шифртекстов дает новый шифртекст, хранящий $m_0 \cdot m'_0,...,m_{l-1} \cdot m'_{l-1} \in \mathbf{K}_n^l$. Смарт и Веркотерен использовали это наблюдение для создания пакетной (или SIMD [12]) системы гомоморфного шифрования.

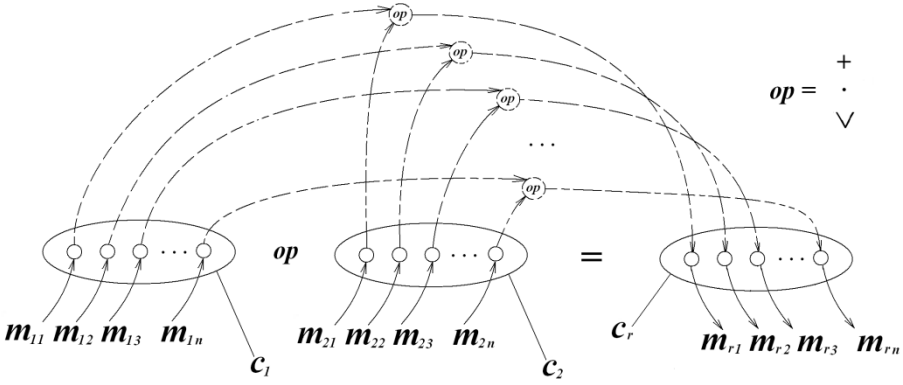


Рис. 1. Выполнение SIMD операций с пакетными шифртекстами.

Говоря о пакетном шифровании и SIMD криптосистемах удобно говорить о составном (Aggregate) пространстве открытых текстов и ключей. Дело в том, что, поскольку над всеми содержащимися в шифртексте открытыми данными

производятся параллельно одни и те же операции, можно рассматривать такие наборы открытых данных как единые элементы пространства наборов открытых данных.

Идея пакетного шифрования получила развитие и использование в работах [6-13] благодаря возможности переставлять открытые тексты внутри одного шифртекста без расшифрования. Это открывает большие перспективы гомоморфной обработки данных, в частности делает возможным проведение над зашифрованными числами в битовом представлении стандартных машинных операций таких как Add, Mult, Xor (т.е. сложение, умножение, деление в битовом представлении, побитовое исключающее или и сравнение). Конкретно, перестановка битов данных между ячейками (слотами, slots) одного шифртекста может быть реализована по-разному, например в [12] для этой цели используется т.н. автоморфизм Фробениуса, а работе [27] описывается использование для этих целей сетей Бенеша.

Пакетное гомоморфное шифрование имеет настолько важное практическое значение, что процедуры для его реализации были включены в недавно вышедшую программную библиотеку HELib компании IBM [28].

4. Построение симметричных гомоморфных SIMD шифров на основе матричных полиномов

Вкратце напомним устройство полностью гомоморфной криптосхемы из [24], основанной на использовании булевых матричных полиномов. Открытыми текстами являются элементы кольца классов вычетов $\mathbf{Z}_p$ по модулю простого числа p, секретный ключ состоит из матрицы $\mathbf{K} \in \mathbf{Z}_p^{N\times N}$ и вектора $\vec{k} \in \mathbf{Z}_p^N$. Открытый текст $m \in \mathbf{Z}_p$ сначала кодируется в матрицу $\mathbf{M} \in \mathbf{Z}_p^{N\times N}$, такую что $\mathbf{M}\cdot\vec{k} = m\cdot\vec{k}$ и $\mathbf{M} \in Comm(\mathbf{K})$, а затем в матричный полином $\mathbf{C}(X) = \mathbf{R}(X)\cdot(X-\mathbf{K}) + \mathbf{M}$, где $\mathbf{R}(X)$ – случайный матричный полином. После умножения двух таких шифртекстов результат приводится по модулю матричного полинома вида $\hat{\mathbf{R}}(X)\cdot(X-\mathbf{K})$, называемого ключом перешифрования. Семантическая криптостойкость такого шифра связана с задачей нахождения корней булевых матричных полиномов [29].

Определение 4. (Задача нахождения корней булевого матричного полинома) Экземпляр (N,d,n)-задачи нахождения корней булевого матричного полинома состоит в том, чтобы по заданному матричному полиному $\mathbf{F}(X)$ степени d с коэффициентами из матричного кольца $\mathbf{Z}_n^{N\times N}$, ответить на вопрос есть ли корни у матричного полинома (распознавательный вариант задачи) и найти эти корни (вычислительный вариант задачи).

Применительно к матричным полиномам концепция SIMD может быть реализована как минимум тремя способами:

1) с использованием китайской теоремы об остатках;
2) путем записи в одной матрице нескольких различных собственных значений при различных собственных векторах;
3) с помощью интерполяции матричных полиномов.

Рассмотрим эти концепции по порядку. Использование китайской теоремы об остатках в духе [9] – наиболее перспективный путь, однако он требует построения обширной алгебраической теории. Использование нескольких собственных чисел матрицы – простой, но не очень эффективный путь. Рассмотрим далее реализацию пакетного шифрования с помощью интерполяции матричных полиномов.

Теорема 2 (Об интерполяции матричных полиномов) Для заданных m пар матриц $(\mathbf{X}_i, \mathbf{Y}_i), i = 1,...,m$ существует матричный полином $\mathbf{A}(X) = \mathbf{A}_m \cdot X^m + \mathbf{A}_{m-1} \cdot X^{m-1} + ... + \mathbf{A}_1 \cdot X + \mathbf{A}_0$ такой, что $\mathbf{A}(\mathbf{X}_i) = \mathbf{Y}_i, i = 1,...,m$ в случае если блочно-матричная система линейных уравнений

$$\left(\mathbf{A}_1, \mathbf{A}_2, ..., \mathbf{A}_m\right) \cdot \begin{pmatrix} \mathbf{I} & \mathbf{I} & ... & \mathbf{I} \\ X_1 & X_2 & ... & X_m \\ ... & ... & ... & ... \\ X_1^{m-1} & X_2^{m-1} & ... & \end{pmatrix} = \left(\mathbf{Y}_1, \mathbf{Y}_2, ..., \mathbf{Y}_m\right) \qquad (3)$$

имеет решение.

В нижеописанной криптосхеме (составным) пространством открытых текстов является $\mathbf{Z}_p^l$ (пространство l-мерных векторов с элементами из $\mathbf{Z}_p$), пространством шифртекстов – $\mathbf{Z}_p^{N \times N}[X]$ (кольцо матричных полиномов), (составным) пространством ключей – вектор пар «матрица из $\mathbf{Z}_p^{N \times N}$, вектор из $\mathbf{Z}_p^N$» вместе с некоторой обратимой матрицей из $\mathbf{Z}_p^{N \times N}$. Опишем далее алгоритмы криптосхемы.

Алгоритм 1. Генерация ключа (KeyGen)

Входные данные: параметр уровня криптостойкости λ, модуль пространства открытых текстов p, количество ячеек l.

Результат: секретный ключ **sk**, ключ перешифрования **rk**.

1. Установить $N \leftarrow \lambda, d \leftarrow \omega(\lambda)$.
2. Выбрать произвольную обратимую матрицу $\mathbf{K}_0 \in GL(N, \mathbf{Z}_p)$.
3. Выбрать l матриц $\mathbf{K}_i, i = 1,...,l$, таких, что $\mathbf{K}_i \neq \mathbf{K}_i^2 \neq ... \neq \mathbf{K}_i^{p-1}$, $\mathbf{I} \in Sp(\mathbf{K}_i)$.
4. Для каждой матрицы $\mathbf{K}_i$ выбрать случайный собственный вектор $\vec{k}_i$.
5. $\mathbf{sk} \leftarrow \{(\mathbf{K}_i, k_i), i = 1,...,l\}, \mathbf{K}_0$
6. Сгенерировать случайный приведенный матричный полином $\hat{\mathbf{R}}(X), \deg\left(\hat{\mathbf{R}}(X)\right) \leq d$.
7. Сгенерировать приведенный матричный полином, $\mathbf{S}(X), \deg\left(\mathbf{S}(X)\right) = l + 1$ такой что $\mathbf{S}(\mathbf{K}_i) = \mathbf{0}, i = 1,...,l$ (т.е. все $\mathbf{K}_i$ – его корни).
8. $\mathbf{R}(X) \leftarrow \hat{\mathbf{R}}(X) \cdot \mathbf{S}(X)$
9. $\mathbf{rk} \leftarrow \mathbf{R}(X - \mathbf{K}_0)$

Замечание 1: Матрицы, удовлетворяющие условию $\mathbf{K}_i \neq \mathbf{K}_i^2 \neq ... \neq \mathbf{K}_i^{p-1}$ нужны для эффективной генерации матриц из $Comm\left(\mathbf{K}_i\right)$, поскольку известно что линейные комбинации степеней матрицы гарантированно лежат в её коммутанте. В наиболее важном случае $p = 2$ это условие сводится к $\mathbf{K}_i \neq \mathbf{K}_i^2$, такие матрицы называются *неидемпотентными*.

Замечание 2: Наличие единицы в спектре матрицы вместе с предыдущим условием является гарантией возможности выбора из $Comm\left(\mathbf{K}_i\right)$ нетривиальных матриц с произвольными собственными числами.

Замечание 3: Запись $\omega(\lambda)$ обозначает некоторую функцию, линейную от λ (т.е. $\omega(\lambda) = \mathrm{O}(\lambda)$), её конкретизация существенна для анализа криптостойкости, но несущественна для анализа асимптотической сложности вычислений над шифртекстами.

Алгоритм 2. Зашифрование данных (Encrypt)

Входные данные: вектор-сообщение открытых текстов $m_i, i = 1,...,l$, секретный ключ **sk.**

Результат: матричный полином шифртекста.

1. Для каждого $m_i, i = 1,...,l$ выбрать случайную матрицу $\mathbf{M}_i$ такую что m_i будет являться её собственным числом при собственном векторе $\vec{k}_i$.
2. С помощью алгоритма интерполяции матричных многочленов вычислить $\hat{\mathbf{C}}(X)$ такой, что $\deg\left(\hat{\mathbf{C}}(X)\right) \le N + \omega(\lambda)$, $\hat{\mathbf{C}}(\mathbf{K}_i) = \mathbf{M}_i$.
3. Вычислить $\mathbf{C}(X) = \hat{\mathbf{C}}(X - \mathbf{K}_0)$.
4. Вернуть в качестве результата $\mathbf{C}(X)$.

Умножение матричных полиномов в Алгоритме 2 может быть выполнено как по определению (формулам (1)), так и с использованием более эффективных алгоритмов, что будет рассмотрено далее.

Алгоритм 3. Расшифрование (Decrypt)

Входные данные: Матричный полином шифртекста $\mathbf{C}(X)$, секретный ключ **sk.**

Результат: сообщение открытого текста $m \in \mathbf{Z}_p$.

1. $\hat{\mathbf{C}}(X) \leftarrow \mathbf{C}(X - \mathbf{K}_0^{-1})$
2. Для каждого $i = 1,...,l$ выполнить $\mathbf{M}_i \leftarrow \hat{\mathbf{C}}(\mathbf{K}_i)$, для ненулевой координаты $(k_j^{-1})_i$ вектора k_i вычислить $m_i = (k_j^{-1})_i \left(\mathbf{M}_i \cdot \vec{k}_i\right)$.
3. Вернуть в качестве результата $\left(m_1,...,m_l\right)$.

Криптосхема поддерживает как аддитивный, так и мультипликативный гомоморфизмы. После умножения двух шифртекстов для понижения степени результат нужно приводить по модулю ключа перешифрования – матричного полинома. Деление можно выполнять, например с помощью алгоритма, указанного в [25].

Корректность расшифрования основывается на обобщенной теореме Безу (для матричных полиномов).

Теорема 2 (Обобщенная теорема Безу) Если S – корень матричного полинома $\mathbf{M}(X)$, то справедливо

$$M(\lambda) = Q(\lambda) \cdot (I\lambda - S),$$

где $Q(\lambda)$ – матричный полином степени $m-1$.

Доказательство теоремы приведено в [30,31] для матричных полиномов над комплексными числами, однако оно справедливо и для матричных полиномов над конечными полями (требование алгебраической замкнутости поля в доказательстве не используется)

Лемма 2 (корректность расшифрования) *Расшифрование вышеописанной криптосхемы корректно и является гомоморфизмом для всех арифметических схем, состоящих из сложений и умножений по модулю* p.

Для обоснования корректности расшифрования достаточно заметить, что подстановка полинома указанного вида – изоморфизм колец.

Лемма 3. *Вышеописанная криптосхема компактна.*

Для обоснования этого утверждения достаточно заметить что в процессе вычислений над шифртекстами степень матричных полиномов результата не превысит заданной.

Анализ сложности умножения шифртекстов. Самой значимой характеристикой эффективности ПГШ является анализ сложности алгоритма произведения двух шифртекстов. По соображениям, сходным с описанными в [24] и [25], асимптотическая сложность этой операции составит $\approx O(\lambda^{3.76})$.

Важный вопрос о криптостойкости будет освещен в расширенном варианте статьи, однако стоит отметить, что при выполнении криптоанализа подобного выполненному в работах [24] и [25] видно что вышеописанная криптосхема может иметь достаточно высокую криптостойкость.

5. Результаты экспериментов и сравнение с аналогами

Для оценки производительности полученных криптосистем автором была сделана тестовая реализация вышеописанных алгоритмов с помощью библиотеки NTL в среде программирования Qt Creator 1.3.1. Для тестирования использовался ноутбук с процессором AMD Phenon 1.8 Quad Core 2 с оперативной памятью 4 Гб. При реализации были использованы следующие параметры: открытые тексты выбираются из $\mathbf{Z}_2$, количество открытых текстов на один шифртекст – 11 , степень матричного полинома – 12. Таким образом, на каждый бит открытого текста приходится приблизительно 157 битов шифртекста при 144-битной криптостойкости. Время, необходимое для умножения двух шифртекстов – 50 мсек.

В статье [26] исследователи из IBM Крейг Джентри, Дэн Боне и соавт. представили реализацию криптосхемы из [8] со следующими параметрами: уровень криптостойкости – 128 бит, количество открытых текстов на один шифртекст – 7866, модуль пространства открытых текстов $p = 1000021573$, $\log_2 q = 238$. На каждый бит открытого текста в таком случае приходится приблизительно 218 битов шифртекста. В [32] приводятся следующие данные по производительности: на Intel Core i7-2600 с 3.4 ГГц и более 200 Гб ОЗУ умножение двух шифртекстов при 128 битной криптостойкости выполняется за 148 мсек.

6. Заключение

Были описаны и проанализированы возможные подходы к построению пакетного ПГШ на основе матричных полиномов, а также представлен набор алгоритмов, реализующий один из этих подходов – криптосхему ПГШ с интерполяцией матричных полиномов. Было показано, что по эффективности построенная криптосхема превосходит аналоги, разработанные исследователями из IBM. Более полное описание криптосхем с обоснованием криптостойкости будет приведено в отдельной статье.

Список литературы

[1]. *R. L. Rivest, L. Adleman, M. L. Dertouzos*. On data banks and privacy homomorphisms. *Foundations of secure computation*. 1978, Т. 4. №. 11. pp. 169-180.

[2]. *C. Gentry*. Fully homomorphic encryption using ideal lattices. *Proceedings of the 41st annual ACM symposium on Symposium on theory of computing-STOC'09*. Vol. 9 – ACM Press, 2009. pp. 169-169. doi:10.1145/1536414.1536440

[3]. *A. Silverberg*. Fully homomorphic encryption for mathematicians. *Women in Numbers 2: Research Directions in Number Theory*. – 2013. – Т. 606. – pp. 111.

[4]. *Н. П. Варновский, А. В. Шокуров*. Гомоморфное шифрование. *Труды ИСП РАН*, том 12, 2007 г. стр. 27-36.

[5]. *А. О. Жиров, О. В. Жирова, С. Ф. Кренделев*. Безопасные облачные вычисления с помощью гомоморфной криптографии. *Журнал БИТ (безопасность информационных технологий)*, том 1, 2013. стр. 6–12.

[6]. *Nigel P. Smart, F. Vercauteren*. Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. *Public Key Cryptography-PKC 2010: 13th International Conference on Practice and Theory in Public Key Cryptography, Paris, France, May 26-28, 2010, Proceedings*. – Springer, 2010. p. 420.

[7]. *M. Naehrig, K. Lauter, V. Vaikuntanathan*. Can homomorphic encryption be practical? *Proceedings of the 3rd ACM workshop on Cloud computing security workshop. – ACM*, 2011. pp. 113-124. doi: 10.1145/2046660.2046682

[8]. *C. Gentry, S. Halevi, N. P. Smart*. Fully homomorphic encryption with polylog overhead. *Advances in Cryptology–EUROCRYPT 2012. – Springer Berlin Heidelberg*, 2012. pp. 465-482. doi: 10.1007/978-3-642-29011-4_28

[9]. *J. H. Cheon, J. S. Coron, J. Kim, M. S. Lee, T. Lepoint, M. Tibouchi, A. Yun.* Batch Fully Homomorphic Encryption over the Integers. Advances in Cryptology–EUROCRYPT 2013. – Т. 7881. – 2013. pp. 315-335. doi: 10.1007/978-3-642-38348-9_20

[10]. *Z. Brakerski, C. Gentry, S. Halevi*. Packed ciphertexts in LWE-based homomorphic encryption. *Public-Key Cryptography–PKC 2013. – Springer Berlin Heidelberg*, 2013. – pp. 1-13. doi: 10.1007/978-3-642-36362-7_1

[11]. *M. Yasuda, T. Shimoyama, J. Kogure, K. Yokoyama, T. Koshiba*. Packed homomorphic encryption based on ideal lattices and its application to biometrics. *Security Engineering and Intelligence Informatics. – Springer Berlin Heidelberg, 2013*. pp. 55-74.

[12]. *Nigel P. Smart, F. Vercauteren*. Fully homomorphic SIMD operations. *Designs, codes and cryptography*, 2014. – Т. 71. – №. 1. – pp. 57-81. doi: 10.1007/s10623-012-9720-4

[13]. *M. Yasuda, T. Shimoyama, J. Kogure, K. Yokoyama, T. Koshiba*. Practical packing method in somewhat homomorphic encryption. *Data Privacy Management and Autonomous Spontaneous Security. – Springer Berlin Heidelberg*, 2014. pp. 34-50.

[14]. *A. Zhirov, O. Zhirova, S. F. Krendelev*. Practical fully homomorphic encryption over polynomial quotient rings. *Internet Security (WorldCIS), 2013 World Congress on. – IEEE*, 2013. pp. 70-75. doi: 10.1109/WorldCIS.2013.6751020

[15]. *M. Hojsík, V. Půlpánová*. A fully homomorphic cryptosystem with approximate perfect secrecy. *Proceedings of the 13th international conference on Topics in Cryptology. – Springer-Verlag*, 2013. pp. 375-388. doi: 10.1007/978-3-642-36095-4_24

[16]. *J. Domingo-Ferrer*. A Provably Secure Additive and Multiplicative Privacy Homomorphism*. *Information Security. – Springer Berlin Heidelberg*, 2002. pp. 471-483.

[17]. *G. Gavin*. An efficient FHE based on the hardness of solving systems of non-linear multivariate equations. *IACR Cryptology ePrint Archive*, 2013. №. 262.

[18]. *M. R. Albrecht, P. Farshim, J. C. Faugere, L. Perret*. Polly cracker, revisited. *Advances in Cryptology–ASIACRYPT 2011. – Springer Berlin Heidelberg*, 2011. pp. 179-196.

[19]. *G. Herold*. Polly cracker, revisited, revisited. *Public Key Cryptography–PKC 2012. – Springer Berlin Heidelberg*, 2012. – pp. 17-33.

[20]. *F. Armknecht, D. Augot, L. Perret, A. R. Sadeghi*. On constructing homomorphic encryption schemes from coding theory. *Cryptography and Coding. – Springer Berlin Heidelberg*, 2011. – pp. 23-40.

[21]. *А. Г. Ростовцев, А. Богданов, М. Михайлов*. Метод безопасного вычисления полинома в недоверенной среде с помощью гомоморфизмов колец. *Проблемы информационной безопасности. Компьютерные системы*, том 2, 2011, стр. 76-85.

[22]. *D. Wagner*. Cryptanalysis of an algebraic privacy homomorphism. *Proc. of 6th Information Security Conference (ISC'03)*. – 2003. doi: 10.1.1.5.1420

[23]. *A. Trepacheva, L. Babenko*. Known plaintexts attack on polynomial based homomorphic encryption. *Proceedings of the 7th International Conference on Security of Information and Networks. – ACM*, 2014. – pp. 157. doi: 10.1145/2659651.2659692

[24]. *Ph. Burtyka, O. Makarevich*. Symmetric Fully Homomorphic Encryption Using Decidable Matrix Equations. *Proceedings of the 7th International Conference on Security of Information and Networks. ACM*, 2014, pp. 186–196. doi: 10.1145/2659651.2659693

[25]. *Ф. Б. Буртыка*. Симметричное полностью гомоморфное шифрование с использованием неприводимых матричных полиномов. *Известия Южного федерального университета. Технические науки*, том 158, №. 9, стр. 107-122, 2014.

[26]. *D. Boneh, C. Gentry, S. Halevi, F. Wang, D. J. Wu*. Private database queries using somewhat homomorphic encryption. *Applied Cryptography and Network Security. – Springer Berlin Heidelberg*, 2013. – pp. 102-118. doi: 10.1007/978-3-642-38980-1_7

[27]. *S. Halevi, V. Shoup*. Algorithms in HElib. *IACR Cryptology ePrint Archive*, 2014. № 106.

[28]. *S. Halevi*. (2012) Performance of HElib. [Online]. Available: http://mpclounge.files.wordpress.com/2013/04/hespeed.pdf (Дата обращения 18.12.2014)

[29]. *Ф. Б. Буртыка*. О сложности нахождения корней булевых матричных полиномов. *Математическое моделирование*, том 27, 2015. – №. 7.

[30]. *Jr J. E. Dennis, J. F. Traub, R. P. Weber*. The algebraic theory of matrix polynomials. *SIAM Journal on Numerical Analysis*, 13(6), 1976. pp. 831-845.

[31]. *Jr J. E. Dennis, J. F. Traub, R. P. Weber*. Algorithms for solvents of matrix polynomials. *SIAM Journal on Numerical Analysis*, 1978. – Т. 15. – №. 3. – pp. 523-533.

[32]. *Antoine Guellier*. Can Homomorphic Cryptography ensure Privacy? [Research Report] RR-8568, 2014, pp.111. https://hal.inria.fr/hal-01052509v1

Batch Symmetric Fully Homomorphic Encryption Using Matrix Polynomials

Ph. Burtyka <bbfilipp@ya.ru>
Southern Federal University,
105/42, Bolshaya Sadovaya st., Rostov-on-Don, 344006, Russia

Abstract. Fully homomorphic encryption (FHE) is a recognized tool to obtain the cryptographic protection of cloud computing. However, the characteristics of existing FHE schemes are not sufficient for use in practice – the security of some FHE is unsatisfying, others require too much computational resources. For improvement the efficiency of the last one IBM researchers proposed a method for "ciphertexts batching", which was applied by them to public key FHE scheme whose security is based on the complexity of the lattice theory hardness assumptions. In this paper, we discuss several methods for embedding "ciphertexts batching" into recently proposed symmetric encryption scheme based on matrix polynomials. For one of this method we completely specify how cryptosystem algorithms should work. The results of computer experiments are given.

Keywords: information security, cloud computing, fully homomorphic encryption, batch encryption, matrix polynomials, secret computations.

References

[1]. *R. L. Rivest, L. Adleman, M. L. Dertouzos*. On data banks and privacy homomorphisms. *Foundations of secure computation*. 1978, Vol. 4, №. 11. pp. 169-180

[2]. *C. Gentry*. Fully homomorphic encryption using ideal lattices. *Proceedings of the 41st annual ACM symposium on Symposium on theory of computing-STOC'09. – ACM Press*, 2009. Vol. 9, pp. 169-169. doi:10.1145/1536414.1536440

[3]. *A. Silverberg*. Fully homomorphic encryption for mathematicians. *Women in Numbers 2: Research Directions in Number Theory*, 2013. Vol. 606. p. 111.

[4]. *N.P. Varnovskij, A.V. Shokurov*. Gomomorfnoe shifrovanie. [Homomorphic encryption]. *Trudy ISP RAN [The Proceedings of ISP RAS]*, 2007. Vol. 12, pp. 27-36. (in Russian).

[5]. *O. Zhirov, O. V. Zhirova, and S. F. Krendelev*. Bezopasnye oblachnye vychisleniya s pomoshhyu homomorfnoj cryptographii. [Secure cloud computing using homomorphic cryptography]. *BIT (bezopasnost' informacionnyx technology) journal [Security of Information Technologies Magazine]*, 2013, Vol. 1, pp. 6–12. (in Russian).

[6]. *Nigel P. Smart, F. Vercauteren*. Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. *Public Key Cryptography-PKC 2010: 13th International Conference on Practice and Theory in Public Key Cryptography, Paris, France, May 26-28, 2010, Proceedings. – Springer*, 2010. p. 420.

[7]. *Naehrig M., Lauter K., Vaikuntanathan V*. Can homomorphic encryption be practical? *Proceedings of the 3rd ACM workshop on Cloud computing security workshop. – ACM*, 2011. pp. 113-124. doi: 10.1145/2046660.2046682

[8]. *C. Gentry, S. Halevi, N. P. Smart*. Fully homomorphic encryption with polylog overhead. *Advances in Cryptology–EUROCRYPT 2012. – Springer Berlin Heidelberg*, 2012. pp. 465-482. doi: 10.1007/978-3-642-29011-4_28

[9]. *Cheon, J. H., Coron, J. S., Kim, J., Lee, M. S., Lepoint, T., Tibouchi, M., Yun, A*. Batch Fully Homomorphic Encryption over the Integers. *Advances in Cryptology–EUROCRYPT*. – Vol. 7881. – 2013. pp. 315-335. doi: 10.1007/978-3-642-38348-9_20

[10]. *Z. Brakerski, C. Gentry, S. Halevi*. Packed ciphertexts in LWE-based homomorphic encryption. *Public-Key Cryptography–PKC 2013. – Springer Berlin Heidelberg, 2013*. – pp. 1-13. doi: 10.1007/978-3-642-36362-7_1

[11]. *Yasuda, M., Shimoyama, T., Kogure, J., Yokoyama, K., Koshiba, T*. Packed homomorphic encryption based on ideal lattices and its application to biometrics. *Security Engineering and Intelligence Informatics. – Springer Berlin Heidelberg*, 2013. pp. 55-74.

[12]. *Nigel P. Smart, F. Vercauteren*. Fully homomorphic SIMD operations. *Designs, codes and cryptography*, 2014. Vol. 71, №. 1. – pp. 57-81. doi: 10.1007/s10623-012-9720-4

[13]. *Yasuda, M., Shimoyama, T., Kogure, J., Yokoyama, K., Koshiba, T*. Practical packing method in somewhat homomorphic encryption. *Data Privacy Management and Autonomous Spontaneous Security. – Springer Berlin Heidelberg*, 2014. pp. 34-50.

[14]. *Zhirov A., Zhirova O., Krendelev S*. F. Practical fully homomorphic encryption over polynomial quotient rings. *Internet Security (WorldCIS), 2013 World Congress on. – IEEE*, 2013. pp. 70-75. doi: 10.1109/WorldCIS.2013.6751020

[15]. *Hojsík M., Půlpánová V*. A fully homomorphic cryptosystem with approximate perfect secrecy. *Proceedings of the 13th international conference on Topics in Cryptology. – Springer-Verlag*, 2013. pp. 375-388. doi: 10.1007/978-3-642-36095-4_24

[16]. *J. Domingo-Ferrer*. A Provably Secure Additive and Multiplicative Privacy Homomorphism*. *Information Security. – Springer Berlin Heidelberg*, 2002. pp. 471-483.

[17]. *Gavin G*. An efficient FHE based on the hardness of solving systems of non-linear multivariate equations. *IACR Cryptology ePrint Archive*, 2013. №. 262.

[18]. *Albrecht, M. R., Farshim, P., Faugere, J. C., Perret, L*. Polly cracker, revisited. *Advances in Cryptology–ASIACRYPT 2011. – Springer Berlin Heidelberg*, 2011. pp. 179-196.

[19]. *Herold G*. Polly cracker, revisited, revisited. *Public Key Cryptography–PKC 2012. – Springer Berlin Heidelberg*, 2012. – pp. 17-33.

[20]. *Armknecht, F., Augot, D., Perret, L., Sadeghi, A. R*. On constructing homomorphic encryption schemes from coding theory. *Cryptography and Coding. – Springer Berlin Heidelberg*, 2011. – pp. 23-40.

[21]. *Rostovtsev A., Bogdanov A., Mikhaylov M*. Metod bezopasnogo vychislenija polinoma v nedoverennoj srede s pomowqju gomomorfizmov kolec [Secure evaluation of polynomial using privacy ring homomorphisms]. *Problemy informacionnoj bezopasnosti. Kompqjuternye sistemy* [*Information security issues. Computer systems*], 2011. Vol. 2. – pp. 76-85. (in Russian).

[22]. *Wagner D*. Cryptanalysis of an algebraic privacy homomorphism. *Proc. of 6th Information Security Conference (ISC'03)*. – 2003. doi: 10.1.1.5.1420

[23]. *Trepacheva A., Babenko L*. Known plaintexts attack on polynomial based homomorphic encryption. *Proceedings of the 7th International Conference on Security of Information and Networks. – ACM*, 2014. – pp. 157. doi: 10.1145/2659651.2659692

[24]. *Ph. Burtyka, O. Makarevich*. Symmetric Fully Homomorphic Encryption Using Decidable Matrix Equations. *Proceedings of the 7th International Conference on Security of Information and Networks. ACM*, 2014, pp. 186–196. doi: 10.1145/2659651.2659693

[25]. *F. B. Burtyka*. Simmetrichnoe polnost'ju gomomorfnoe shifrovanie s ispol'zovaniem neprivodimyh matrichnyh polinomov [Symmetric fully homomorphic encryption using irreducible matrix polynomials]. *Izvestija Juzhnogo federal'nogo universiteta. Tehnicheskie nauki [Proceedings of Southern Federal University. Engineering sciences]*, 2014, Vol. 158, №. 9, pp. 107-122. (in Russian).

[26]. *Boneh, D., Gentry, C., Halevi, S., Wang, F., Wu, D. J.* Private database queries using somewhat homomorphic encryption. *Applied Cryptography and Network Security. – Springer Berlin Heidelberg*, 2013. – pp. 102-118. doi: 10.1007/978-3-642-38980-1_7

[27]. *S. Halevi, V. Shoup*. Algorithms in HElib. *IACR Cryptology ePrint Archive*, 2014. №. 106.

[28]. *S. Halevi*. (2012) Performance of HElib. [Online]. Available: http://mpclounge.files.wordpress.com/2013/04/hespeed.pdf (Visited on 18.12.2014)

[29]. *Burtyka Ph. B.* O slozhnosti naxozhdenija kornej bulevyx matrichnyx polinomov [On the complexity of finding the roots of Boolean matrix polynomials]. *Matematicheskoe modelirovanie [Matematical modelling]*, 2015. Vol. 27, №. 7. (in Russian).

[30]. *Dennis, Jr J. E., Traub J. F., Weber R. P*. The algebraic theory of matrix polynomials. *SIAM Journal on Numerical Analysis*, 13(6), 1976. pp. 831-845.

[31]. *Dennis, Jr J. E., Traub J. F., Weber R. P*. Algorithms for solvents of matrix polynomials. *SIAM Journal on Numerical Analysis*, 1978. Vol. 15. – №. 3. – pp. 523-533.

[32]. *Antoine Guellier*. Can Homomorphic Cryptography ensure Privacy? [Research Report] RR-8568, 2014, pp.111. https://hal.inria.fr/hal-01052509v1

Прямое численное моделирование аттракторов внутренних волн стратифицированной жидкости в трапециедальной области с колеблющейся вертикальной стенкой

К. Брузе [1] <christophe.brouzet@ens-lyon.fr >
Т. Доксуа [1] <thierry.dauxois@ens-lyon.fr >
Е. Ерманюк [1,2] <ermanyuk@gmail.com>
С. Жубо [1] <sylvain.joubaud@ens-lyon.fr>
М. Крапошин [3,4] <os-cfd@yandex.ru>
И. Сибгатуллин,[4,5,6] <ilias_s@mail.ru>
[1] *Laboratoire de Physique de l'École Normale Supérieure de Lyon, Universite de Lyon, France*
[2] *Институт гидродинамики им. М.А. Лаврентьева, Новосибирск, Россия*
[3] *Национальный исследовательский центр "Курчатовский институт", Москва, Россия*
[4] *Институт системного программирования РАН, Москва, Россия*
[5] *Механико-математический факультет и институт механики МГУ им. М.В. Ломоносова, Москва, Россия*
[6] *Институт океанологии им. П.П. Ширшова РАН, Москва, Россия*

Список обозначений

s - абсолютная солёность, --

$\vec{U}$ - скорость среды, м/с

ρ - плотность среды, м3/кг

p - статическое давление среды, Па

p^* - пьезометрическое давление, Па

ρ_s - плотность солёной воды, кг/м3

$\vec{g}$ - ускорение свободного падения, м/с2

$\vec{r}$ - радиус-вектор, м

μ - динамическая вязкость, Па-с

ν - кинематическая вязкость, м2/с

β_s - коэффициент объёмного сужения жидкости за счёт солёности, --

Sc — число Шмидта, --

МКО — Метод Конечного Объёма
МСЭ — Метод Спектральных Элементов

Аннотация. Проведено прямое численное моделирование формирования аттрактора внутренних гравитационных волн в стратифицированной жидкости с помощью двух численных алгоритмов. Математическая постановка задачи соответствует экспериментам по возбуждению внутренних волн через вертикальный волнопродуктор в трапецеидальном контейнере с раствором соли. Показано, что двумерные численные расчеты хорошо воспроизводят экспериментальные данные при учете изменения линейного профиля солености у свободной поверхности. При этом амплитуда колебаний в численных расчетах увеличена за счет потерь энергии волнопродуктора в трехмерном контейнере. Несмотря на то, что общий вид аттрактора хорошо воспроизводится как методом спектральных элементов, так и методом конечных объемов, в профилях скоростей имеются отличия у левой границы. Это требует дальнейшего исследования, поскольку такие отличия могут влиять на нелинейную динамику волн при развитии неустойчивостей.

Ключевые слова: аттрактор, инерционные волны, гравитационные волны, прямое численное моделирование**.**

1. Введение

Внутренние гравитационные волны в однородно стратифицированной жидкости при постоянной частоте плавучести могут распространяться под фиксированным углом к вертикали, который определяется только частотой вынуждающей силы [1]. При наличии наклонённой стенки волны могут фокусироваться и это приводит к образованию определённых путей, которые определяются лишь частотой. Эти пути были названы волновыми аттракторами. Необходимо отметить, что к аттракторам в фазовом пространстве из теории динамических систем данное понятие прямого отношения не имеет. Первые эксперименты, подтверждающие существование предсказанных ранее волновых аттракторов описаны в [1]. Существование подобных аттракторов и анализ их структуры, устойчивости и динамики важным для понимания процессов перемешивания в морях и озёрах, а также

для анализа колебаний во вращающихся системах, таких как ядро Земли, поскольку инерционные волны также могут распространяться по аттракторам.

Устойчиво стратифицированное состояние жидкости обладает частотой плавучести, или частотой Брента-Вяйсяля, которая характеризует малые колебания смещённого из исходного состояния элемента жидкости. Возмущения исходного состояния с постоянной частотой, меньшей частоты плавучести, приводят к внутренним волнам, распространяющимся вдоль прямых линий (в приближении постоянной частоты плавучести). Лучи могут отражаться от поверхностей, при этом после отражения лучи по прежнему распространяются под углом, равным по модулю арккосинусу отношения вынуждающей частоты и частоты плавучести. За счёт такого отличия в характере отражения становится возможной фокусировка внутренних волн в замкнутой области при определенной геометрии. Замкнутые пути, возникающие в результате фокусировки, были названы волновыми аттракторами [1]. В частности, в работе [1] было показано, что такая фокусировка возможна, если одна из стенок области наклонена по отношению к вертикали. Также при формировании аттрактора в реальных условиях играет роль баланс фокусировки и вязкости [6]. Анализ двумерного околокритичного отражения слабонелинейных внутренних гравитационных волн от наклоненной границы в однородностратифицированной жидкости приведен в [2,5]. По существу волновые аттракторы являются двумерными, что показано как теоретически, так и экспериментально [1-5].

Помимо теоретических и экспериментальных исследований волновых аттракторов, предпринимались попытки и их численного исследования: в работе [4] с помощью программного кода общей модели циркуляции (MIT general circulation model), использующей метод конечного объёма, исследовались волновые аттракторы, возникающие при гармоническом вертикальном колебании всей области. Проведено сравнение с экспериментом для устойчивого аттрактора. Известно, что методы конечного объёма обладают собственной численной вязкостью и их применение для анализа устойчивости сильно нелинейных режимов на больших временах требует осторожности. Помимо, этого, в [4] численные исследования проводились для числа Шмидта 100, а не 770, как в эксперименте, из-за невозможности разрешения диффузионных масштабов, а на нижней и правой стенке ставилось условие проскальзывания вместо прилипания. Таким образом численная модель не вполне соответствовала эксперименту. Тем не менее, при определённых параметрах удалось воспроизвести общий вид аттрактора. В работах [10,11] авторами проведены предварительные расчеты методом спектральных элементов (МСЭ).

В работе [7] для смежной задачи для инерционных волн в прямоугольной геометрии проведено прямое численное моделирование аттракторов вращающейся жидкости спектральным методом. Спектральный метод позволяет с высокой точностью рассчитывать нелинейные режимы и

предпочтителен по сравнению с конечно объёмным для исследования нелинейных режимов, но его применение в непрямоугольной геометрии с разнородными граничными условиями затруднено.

Таким образом, при проведении и планировании расчетных исследований течений подобного рода мы сталкиваемся со следующей дилеммой:

- либо использовать метод конечного объёма, основными преимуществами которого являются простота реализации расчётной модели и устойчивость, но при этом внести в результаты расчётов ошибку, связанную с высокой численной диффузией самого метода;
- либо использовать метод спектральных элементов, порядок аппроксимации которого намного выше, чем в МКО (2-ой порядок), при этом потеряв гибкость и простоту создания расчётных моделей.

В этом случае хорошим решением может быть тестирование обоих методов на простой геометрии области моделирования с целью выявления характерных различий между МКО и МСЭ. А в следующих расчётных исследованиях использовать МКО и накопленный опыт для минимизации его недостатков, иными словами — численной диффузии. В качестве таких задач могут использоваться экспериментальные исследования, схожие с описываемыми в работах [3,7]. Эффективное применение МКО к задачам распространения внутренних волн малой амплитуды и движения тел в стратифицированных средах было реализовано в [13].

Другой принципиальной сложностью, например как в случае [3,7], является сопоставление с экспериментом расчётных данных. Это связано с некоторой неопределенностью постановки граничных и начальных условий, заданием физических констант — например, частоты плавучести, распределения плотности в начальный момент времени, числа Шмидта, амплитуды колебания поверхности исследуемого пространства и т. д. Более того, при интерпретации различных режимов течения с помощью расчётного анализа необходимо наличие критериев, показывающих границы этих режимов. Поиск таких критериев с помощью численного моделирования связан с обработкой большого количества данных.

Выходом из положения может быть использование специальных методов обработки данных, таких как POD, SVD и пр., которые позволяют вычислить собственные вектора и значения динамической системы и в дальнейшем, понизив размерность, использовать эти результаты для поиска необходимого решения.

С учётом указанных выше вызовов, исследование целесообразно проводить в следующем порядке:

1) Анализ условий экспериментального стенда для последующего численного моделирования;
2) Выбор математической модели;

3) Поиск приближений и упрощений математической модели, сформулированной в п. 2) для реализации численной модели;
4) Реализация численной модели течения методом конечного объёма (МКО) и методом спектральных элементов (МСЭ);
5) Сравнение МКО и МСЭ для выбранных расчётных случаев;
6) Исследование динамических характеристик моделей выбранных режимов матричными методами ;
7) Обобщённый анализ полученных результатов.

2. Физическая постановка задачи и её математическое описание

Рассматривается двумерное течение однофазной несжимаемой вязкой жидкости — раствора соли в поле силы тяжести. В замкнутом объёме жидкости с заданной стратификацией в начальный момент времени, в движение приводится одна из вертикальных стенок (правая) — см. рис. 1-3. Вертикальная стенка «слева» движется в горизонтальном направлении по заданному закону, стенки снизу и справа — неподвижны, горизонтальная граница сверху — поверхность раздела, на которой задаётся условие «проскальзывания».

Раствор соли заключён в трапецеидальной области, рис. 1. Перед началом эксперимента создаётся вертикальная стратификация соли с заданной частотой плавучести. На нижней и боковых поверхностях ставится условие прилипания. Возникающие в ходе эксперимента внутренние волны практически не влияют на форму поверхности, поэтому на верхней границе ставится условия отсутствия касательных напряжений.

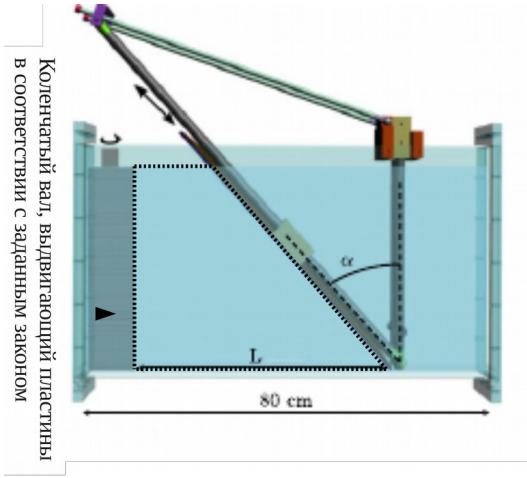


Рисунок 1. Принципиальная схема экспериментальной установки. Жирным пунктиром показана область моделирования

Рисунок 2. Принципиальная схема расчётной области и физических внешних границ

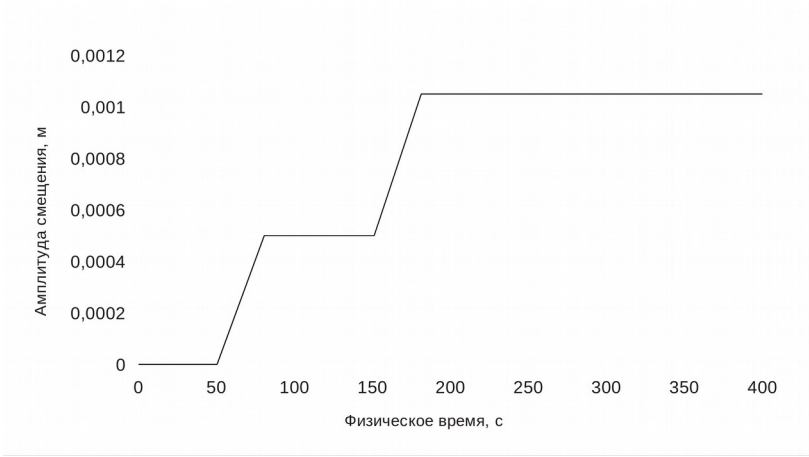


Рисунок 3. Схема «вывода» амплитуды смещения левой вертикальной стенки на максимальное значение.

Для описания расчётной модели используются следующие уравнения сохранения:

уравнение сохранения массы

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \vec{U} \right) = 0 \tag{1}$$

уравнение сохранения импульса

$$\frac{\partial \rho \vec{U}}{\partial t} + \nabla \cdot \left(\rho \vec{U} \otimes \vec{U} \right) - \nabla \cdot \sigma = - \nabla p + \rho \vec{g} \tag{2}$$

уравнение переноса массы растворённой соли

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot \left(\rho_s \vec{U} \right) = \nabla \cdot D_s \nabla \rho_s \tag{3}$$

Слагаемое в правой части уравнения (3) позволяет учесть диффузионные эффекты, которые могут возникать на длительных временах моделирования.

Для учёта влияния массовой концентрации s растворённой соли на плотность жидкости используется выражение:

$$\rho(s) = \rho_0 + \left(\frac{\partial \rho}{\partial s} \right) (s - s_0) \tag{4}$$

Где s — массовая концентрация солёности жидкости — отношение массы соли в элементарном объёме к общей массе среды.

Жидкость является Ньютоновской и касательные напряжения σ подчиняются линейному закону:

$$\sigma = \mu\left(\nabla \vec{U} + (\nabla \vec{U})^T\right) \tag{5}$$

3. Описание расчетной модели

Изменение плотности среды в эксперименте происходит только за счёт изменения концентрации солёности. Поэтому относительное изменение плотности в слое при движении раствора мало. Кроме этого, течение имеет колебательных характер малой амплитуды, то делает относительное изменение плотности при перемещении частицы жидкости ещё меньше. Поэтому в данном эксперименте имеются все предпосылки для использования приближения Буссинеска. Полагая, что изменения плотности за счёт солёности малы, можно воспользоваться приближением Буссинеска. В этом случае можно:

а) пренебречь изменениями плотности в уравнении неразрывности (1);

б) пренебречь изменениями плотности в левой части баланса импульса (2).

Введём следующие величины:

Пьезометрическое давление

$$p^* = p - \rho\, \vec{g}\cdot\vec{r}$$

Коэффициент объёмного сужения жидкости за счёт солёности

$$\beta_s = \frac{1}{v}\left(\frac{\partial v}{\partial s}\right) = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial s}\right)$$

Кинематическая плотность среды $\rho_k = \rho / \rho_0$

Тогда система уравнений (1)-(4) может быть переписана в виде:

$$\nabla\cdot\vec{U} = 0 \tag{6}$$

$$\frac{\partial \vec{U}}{\partial t\rho} + \nabla\times(\vec{U}\otimes\vec{U}) - \nabla\times\left(\left(U\vec{V}\right)(\vec{V})^T\right)\right) - \nabla\frac{p^*}{g\,r} - \frac{1}{\rho}(\vec{r}\times)\nabla_k \tag{7}$$

$$\frac{\partial s}{\partial t} + \nabla\cdot(s\vec{U}) = \nabla\cdot\frac{\nu}{Sc}\nabla s \tag{8}$$

Кинематическая плотность ρ_k вычисляется как $\rho_k = 1-\beta_s(s-s_0)$. В данном исследовании принимается, что величина β_s, также как и кинематическая вязкость ν и число Шмидта считаются заданными для каждого расчётного случая и не меняются.

4. Численная модель с использованием метода конечного объема

Для численной реализации математической модели, описываемой уравнениями (6)-(8) используется метод конечного объёма второго порядка точности по времени и пространству и схема с расщеплением переменных (алгоритм PISO). Использование метода конечного объёма предполагает, что балансные соотношения импульса (7), массы соли (8) и объёма системы (6) должны быть сформулированы в интегральном виде.

Поскольку балансные соотношения составлены для подвижной контрольной массы, необходимо выполнить переход к контрольному объёму. Для этого используется транспортная теорема Рейнольдса (теорема дифференцирования интеграла по переменному объёму) - для любой экстенсивной величины Ψ, характеризуемой интенсивной величиной ψ, содержащейся в контрольной массе $CM(t)$, движущейся со скоростью $\vec{U}_a$, изменение может быть вычислено как сумма изменения Ψ, содержащейся в подвижном объёме $CV(t)$, охватывающем контрольную массу $CM(t)$ в момент времени t плюс относительный поток этой величины $\psi\vec{U}_r d\vec{S} = \psi\left(\vec{U}_a - \vec{U}_b\right)d\vec{S}$ через границы контрольного объёма при его движении относительно контрольной массы:

$$\frac{d\Psi}{dt} = \frac{d}{dt}\int_{CM(t)} \rho\psi dV = \frac{d}{dt}\int_{CV(t)} \rho\psi dV + \int_{aCV(t)} \rho\psi \vec{U}_r \cdot d\vec{S} \quad (9)$$

Или, переписав первое слагаемое правой части с учётом движения границ контрольного объёма $\vec{U}_b$:

$$\frac{d\Psi}{dt} = \frac{d}{dt}\int_{CM(t)} \rho\psi dV = \int_{CV(t)} \frac{\partial \rho\psi}{\partial t} dV + \int_{aCV(t)} \rho\psi \vec{U}_r \cdot d\vec{S} \quad (10)$$

В соответствии с (9) или (10) возможно два подхода:

1) Положить контрольный объём постоянным в пространстве (метод Эйлера) - $\vec{U}_b = 0$. Этот подход удобен либо для случаев, в которых можно либо выделить границы притока и оттока вещества с известными на границе параметрами (импульс, энергия и другие), либо же для тех случаев, когда скорость среды (контрольной массы) направлена по касательной к поверхности контрольного объёма.
2) В тех же случаях, когда скорость движения контрольной массы может быть направлена нормально по отношению к поверхности контрольного объёма, но при этом невозможно выделить выделить поверхности с известными параметрами среды на границе системы, целесообразно принять поверхность контрольного объёма движущейся вместе с поверхностью системы (совмещённый подход Лагранжа-Эйлера).

В соответствии с физической постановкой задачи, в данной работе используется второй подход, поскольку левая стенка исследуемого объёма движется в нормальном направлении. В этом случае уравнения (6)-(8) приводятся к виду (11)-(13):

$$\int_{\partial CV(t)} \vec{U}_a \cdot d\vec{S} = 0 \tag{11}$$

$$\frac{d}{dt}\int_{CV(t)} \vec{U}_a dV + \int_{\partial CV(t)} \vec{U}_a(\vec{U}_r \cdot d\vec{S}) = \int_{\partial CV(t)} \left(\nu(\nabla \vec{U}_a + (\nabla \vec{U}_a)^T)\right)\cdot d\vec{S} - \int_{\partial CV(t)} \frac{p^*}{\rho_0} + (\vec{g}\cdot\vec{r})\rho_k d\vec{S} \tag{12}$$

$$\frac{d}{dt}\int_{CV(t)} s\, dV + \int_{\partial CV(t)} s(\vec{U}_r \cdot d\vec{S}) = \int_{\partial CV(t)} \frac{\nu}{Sc}(\nabla s \cdot d\vec{S}) \tag{13}$$

5. Процедура численного интегрирования уравнений математической модели

В соответствии с выбранным численным методом, был реализован алгоритм интегрирования уравнений (11)-(13). На каждом шаге по времени выполнялись следующие действия:

1) Вычисление нового положения границ расчётной области, обновление геометрии сеточных линий
2) Прогноз поля солёности по имеющимся полям объёмных потоков (полю скорости с предыдущего шага)
3) Обновление поля кинематической плотности

4) Прогноз поля скорости жидкости
5) Решение уравнения для давления
6) Обновление поля объёмных потоков, обновление поля скорости
7) Обновление поля солёности в соответствии с новым полем скорости
8) Переход к новому шагу по времени

При реализации численной модели применялись следующие численные «трюки»:

1) Учёт небаланса импульса на первом шаге по времени

Поскольку в начальный момент времени жидкость предполагается находящейся в состоянии покоя, необходимо выполнение следующего условия (см. (12)):

$$\int_{\partial CV(t)} \frac{p^*}{\rho_0} + (\vec{g}\cdot\vec{r})\rho_k \vec{dS} = 0$$

Это условие можно обеспечить двумя способами: либо найти такое распределение пьезометрического давления, которое в начальный момент времени будет компенсировать столб жидкости, либо же вычесть этот столб перед началом интегрирования:

$$\int_{\partial CV(t)} \frac{p^*}{\rho_0} + (\vec{g}\cdot\vec{r})\rho_k \vec{dS} - \int_{\partial CV(t)} (\vec{g}\cdot\vec{r}(t=0))\rho_k(t=0)\vec{dS}$$

Второй способ является предпочтительным с точки зрения упрощения задания граничных условий,

2) Учёт градиента солёности в уравнении для давления

Чтобы получить уравнение для давления, воспользуемся известной процедурой PISO, представив уравнение сохранения импульса (12) в полудискретном виде

$$V\left(A\vec{U}_a - H\left(\vec{U}_a\right)\right) = -\int_{cv(t)} \nabla \frac{p^*}{\rho_0} dV - \int_{cv(t)} \vec{g}\times\vec{r}\rho\nabla_k g + \vec{r}\times\vec{k}(=0)\rho\nabla_k(=0)dV$$

Где A — вклад в диагональ линеаризованного уравнения сохранения импульса
H — сумма вкладов от недиагональных элементов и источников

Перейдя к среднеинтегральным значениям и сократив на объём V, получаем:

$$\langle \vec{U}_a \rangle = \frac{\langle H(\vec{U}_a) \rangle}{A} - \frac{\langle \nabla \frac{p^*}{\rho_0} \rangle}{A} - \frac{\langle \vec{g} \cdot \vec{r} \nabla \rho_k - \vec{g} \cdot \vec{r}\,(t=0) \nabla \rho_k(t=0) \rangle}{A}$$

После интерполяции на грани ячеек и скалярного умножения на площади граней, получаем выражение для объёмных потоков вещества:

$$\varphi_f = \left(\frac{\langle H(\vec{U}_a) \rangle}{A} - \frac{\langle \nabla \frac{p^*}{\rho_0} \rangle}{A} - \frac{\langle \vec{g} \cdot \vec{r} \nabla \rho_k - \vec{g} \cdot \vec{r}\,(t=0) \nabla \rho_k(t=0) \rangle}{A} \right)_f \cdot \vec{S}_f$$

Представив уравнение неразрывности (11) в дискретном виде через потоки $\varphi_f = (\vec{U}_a)_f \cdot \vec{S}_f$ получаем уравнение для давления:в

$$\sum_f \left(\frac{\langle H(\vec{U}_a) \rangle}{A} - \frac{\langle \vec{g} \cdot \vec{r} \nabla \rho_k - \vec{g} \cdot \vec{r}\,(t=0) \nabla \rho_k(t=0) \rangle}{A} \right)_f \cdot \vec{S}_f = \sum_f \left(\frac{\langle \nabla \frac{p^*}{\rho_0} \rangle}{A} \right)_f \cdot \vec{S}_f$$

Из решения уравнения для давления получаем величину поправки к объёмным потокам, необходимую для точного выполнения уравнения неразрывности (11). Получившаяся схема предполагает хранение поля скорости в центрах контрольных объёмов и потоков, обеспечивающих консервативность — в центрах граней.

3) Граничное условие для давления на внешних границах расчётной области

На границах расчётной области, в которых скорость среды задана постоянной (в данном случае — стенка), следует задать такое граничное условие, которое обеспечит нулевую коррекцию к потокам вещества

$$\int_{\partial CV(t)} \frac{p^*}{\rho_0} + (\vec{g} \cdot \vec{r}) \rho_k \vec{dS} - \int_{\partial CV(t)} (\vec{g} \cdot \vec{r}\,(t=0)) \rho_k(t=0) \vec{dS} = 0$$

4) Движение сеточных линий

К модели течения в неподвижной расчётной области добавляется движение расчётной сетки на каждом шаге перед решением уравнения для солёности. На шаге решения уравнения для давления находятся новые потоки, соответствующие движению левой вертикальной сетки.

Для учёта деформации расчётной области в каждый момент времени решается уравнение Лапласа для смещений граней расчётной области:

$$\int_{CV(t)} \nabla \Gamma_u \nabla \cdot \vec{U}_b \, dV = 0 \qquad (14)$$

6. Описание расчётной области и граничных условий

При разработке расчётной модели и расчётной области (постановке задачи) преследовалась цель максимально возможного соответствия условиям физического эксперимента (см. рис. 1). На рисунке 1 представлена схема экспериментальной установки, на рисунке 2 — соответствующая ей расчётная область. Принято, что поверхность раздела фаз можно считать непроницаемой стенкой, на которой реализуется условие проскальзывания скорости среды. Правая боковая и нижняя стенки непроницаемы и неподвижны, левая боковая стенка движется по синусоидальному закону (см. рис. 2).

Следующим важным предположением является характер выхода на установившийся режим колебаний вертикальной стенки — как следует из рисунка 3, весьма вероятно, что максимальная амплитуда колебаний стенки на начальных этапах процесса (до 200 секунд) изменялась от 0 до номинального значения. Подтвердить это предположение на основе экспериментальных невозможно, поскольку датчики ускорения на стенке отсутствовали. Косвенным подтверждением может быть сравнение режимов с постепенным выходом на номинальную амплитуду и мгновенным приложением смещений в нулевой момент времени. В соответствии с этим, в расчётах амплитуда колебаний стенки изменялась от 0 до номинального значения по ступенчатому закону, показанному на рисунке 3. Данный закон был подобран на основе экспериментальных данных о динамике горизонтальной составляющей скорости в точке отбора (см. рис. 4).

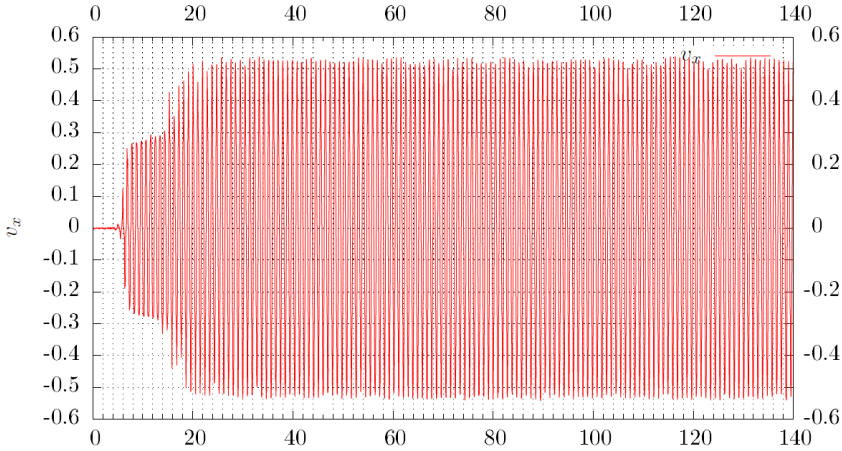


Рис. 4 Экспериментальные данных о динамике горизонтальной составляющей скорости в точке отбора, по горизонтали отложены периоды колебаний левой стенки.

Далее, несмотря на двумерность течения, обнаруженную в эксперименте, использование двумерной модели может также вносить существенный вклад в различие между натурным стендом и расчётной моделью. В данной работе, с целью сокращения вычислительного времени, было решено использовать двумерное приближение. В дальнейшем этот результат может быть уточнён.

Граничные и начальные условия

Граничные и начальные условия задавались в соответствии с постановкой задачи.

Пьезометрическое Давление:

На левой, нижней и правой стенках и верхней стенках — условие непроницаемости.

Скорость:

На левой, нижней и верхней стенках — условие прилипания. На верхней — условие проскальзывания.

Солёность:

На всех стенках условие непротекания (нулевой градиент)

Скорость смещения расчётной области:

на левой вертикальной стенке — переменное значение горизонтальной составляющей в соответствии с заданным законом:

$$U_h(v,t) = a\cos(\pi v/H)(-\omega)\sin(\omega_0 t)$$,

на правой вертикальной стенке — постоянное нулевое значение
на верхней и нижней — нулевой градиент

Начальные условия

Скорость — жидкость находится в покое (все компоненты равны 0)

Пьезометрическое давление — слои жидкости находятся в равновесии — равно давлению на верхней границе

Скорость смещения сеточных линий — все компоненты поля скорости смещения равны нулю

Солёность — распределена по заданному линейному закону от S0 на верхней границе до максимального значения на нижней границе

$$N^2/g = -\beta_s \frac{\partial s_A}{\partial z} = \frac{1}{\rho_0}\frac{\partial \rho}{\partial z}$$

Вместо абсолютной солёности воспользуемся относительной, которая меняется от линейно от 0 (верхняя граница) до 1 (нижняя граница).

Задавшись плотностью жидкости на верхней плоскости, скажем 1000кг/м3, вычисляем плотность на нижней поверхности. Вычисляем значение коэффициента β_s .

N = 1.059 рад/с

g=9.81

$$\rho_0 = 1000$$

Соответствующий градиент плотности

$$\rho_0 N^2/g = 114.32018$$

$$\Delta z = 0.3$$

$$\Delta \rho = 34.296$$

$$\beta_s = -\frac{\Delta \rho}{\Delta s_A}\frac{1}{\rho_0} = -34.296\times 10^{-3}$$

Угловая частота 0.623 рад/с

Амплитуда колебаний 0.15см (1.5мм). Как показали дальнейшие расчётные исследования, величина 0.15см слишком большая для получения устойчивого аттрактора на промежутке времени 0-1500с, поэтому расчётным путём была подобрана амплитуда 0.105см

Положение точки отбора (от нижнего левого угла): +28.2см, +20см

7. Описание расчетной модели методом спектральных элементов.

При описании расчетной модели с использованием метода спектральных элементов [9,12] удобнее пользоваться дифференциальной формой законов сохранения. Уравнения сохранения импульса и диффузии соли в приближении Буссинеска представляются в виде:

$$\left(\frac{\partial \vec{v}}{\partial t} + v^k \nabla_k \vec{v}\right) = -\nabla \frac{p}{\rho_m} + \nu \Delta \vec{v} + s\vec{g} \tag{15}$$

$$\left(\frac{\partial s}{\partial t} + v^k \nabla_k s\right) = \lambda \Delta s, \tag{16}$$

$$\mathrm{div}(v) = 0 \tag{17}$$

Поскольку левая граница совершает гармонические колебания вида

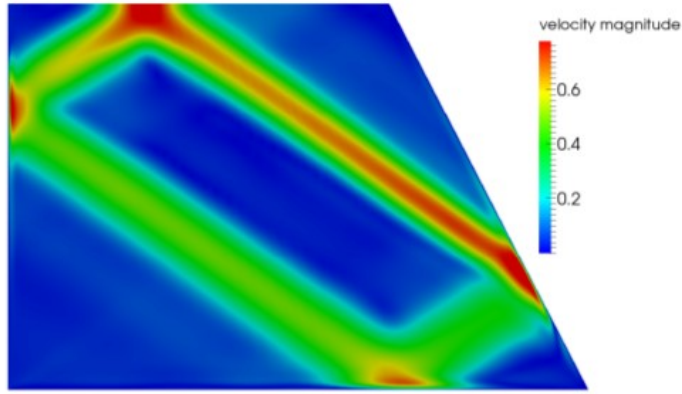
$$x_b(0, y, t) = a\ \cos(\pi y / H)\cos(\omega_0 t),$$

граничные условия в силу малости амплитуды можно записать

$$u(0, y, t) = a\omega_0 \cos(\pi y / H)\sin(\omega_0 t), v = 0$$

Остальные параметры аналогичны параметрам при описании численной модели методом конечного объема. Спектрально-элементный подход сочетает свойства конечно-элементного и спектрального методов. Расчетная область разбивается на элементы, в которых уравнения сохранения дискретизируются с помощью вариационного подхода, по сути являющимся Галеркинским методом. Для разнородных граничных условий удобно использовать базисные функции Лежандра при пространственной дискретизации элементов Гауса-Лобатто [9,12].

При расчетах методом спектральных элементов визуально профиль скорости аналогичен профилю скорости, полученному методом конечного объема:



$$a = 0.15\,cm, N = 1.059 ras/s, \omega_0 = 0.623 rad/s$$

Рисунок 5. Поле модуля скорости расчетов с помощью метода спектральных элементов

В экспериментальной установке трудно добиться идеальной стратификации у верхней границы и волны в основном отражаются от линии, расположенной ниже свободной поверхности. Учет этого обстоятельства при численном расчете приводит к совпадению формы аттрактора в эксперименте и расчете. Ниже показана форма аттрактора при изменении высоты контейнера с 30 см на 28 см и временная зависимость горизонтальной компоненты скорости на луче аттрактора

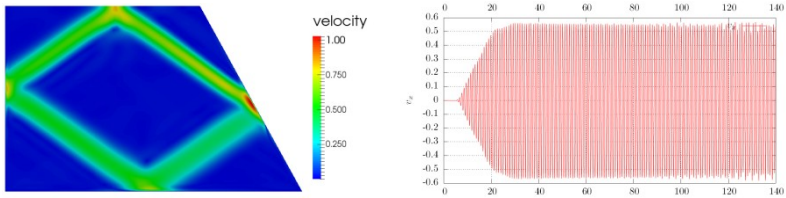


Рисунок 6. Поле модуля скорости и значение горизонтальной составляющей скорости в точке $x = 28.2\,cm\,, y = 20\,cm\,.$

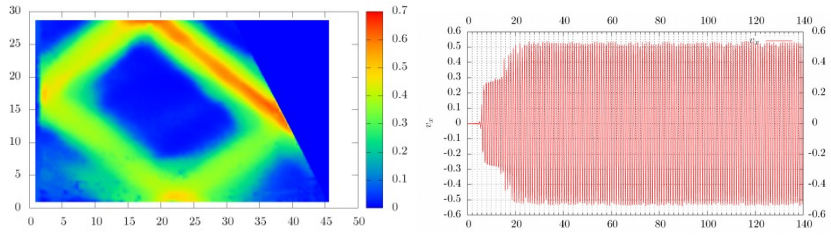


Рисунок 7. Экспериментальное распределение поля модуля скорости, полученное авторами [3] (слева) и соответствующее колебание горизонтальной компоненты скорости в точке $x = 28.2\,см$, $y = 20\,см$ *(справа). По горизонтали отложены периоды вынужденных колебаний.*

8. Результаты тестирования моделей

В соответствии с поставленной задачей выполнено тестирование расчётной модели с использованием метода конечного объёма и метода спектральных элементов. Список режимов показан в таблице 1. Динамика горизонтальной скорости как наиболее показательный параметр для сравнения с экспериментом для базового случая показана на рис. 8, результаты тестирования сеточной сходимости — на рис. 9 и 10. На рис. 11 показано поведение аттрактора (горизонтальной скорости жидкости в выбранной точке) без учёта условий эксперимента. На рисунках 12-13 — распределение вертикальной и соответственно горизонтальной компоненты скорости вдоль линии X=28.2см, на рис. 14-15 — поле модуля скорости. На рисунке 16 показаны точки времени, для которых выполнялось сравнялось МКО и МСЭ. Результаты сравнения МКО и МСЭ представлены на рис. 17. На рис. 18 показано изменение поля горизонтальной компоненты скорости среды во времени в соответствии с рис. 12-13.

Таблица 1. Список режимов для кросс-тестирования МКО и МСЭ

№№	Описание
1	Базовый расчётный случай (соответствует описанию в разделе), сетка 225x150
2	Как случай №1, но с сеткой 450x300
3	Как случай №2, но с шагом по времени 2.5E-3
4	Как случай №3, но с сеткой 900x600
5	Как случай №1, но с отрицательной амплитудой (-0.105см) Для этого случая выводится распределение горизонтальной и вертикальной компонент скорости на линии y=20см от нижнего

	края
6	Как случай №1, но с постоянной амплитудой (0.105см)
7	Как случай №1, но с постоянной амплитудой (-0.105см)

Все расчёты проводились с использованием вычислительных мощностей кластера платформы UniHUB ИСП РАН (http://www.unihub.ru) и суперкомпьютера Ломоносов.

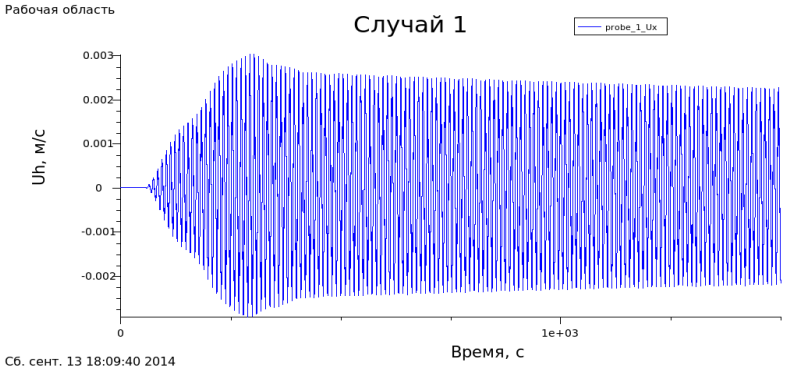


Рисунок 8. Динамика горизонтальной составляющей скорости в точке (28.2, 20) для базового случая (случай 1)

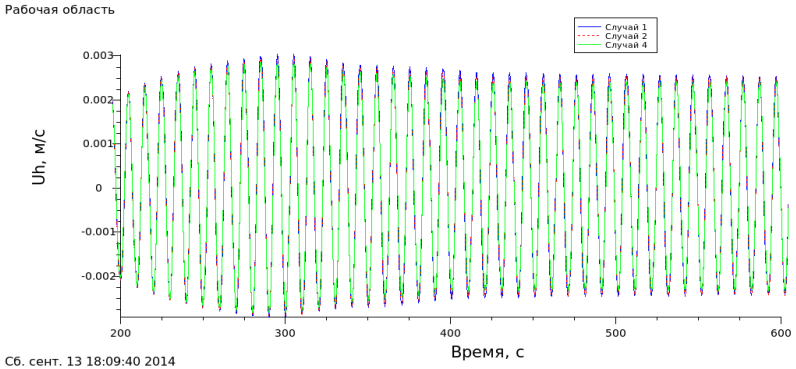


Рисунок 9. Сравнение динамики горизонтальной составляющей скорости среды в точке (28.2, 20) для трёх различных сеточных разрешений (225x150, 450x300, 900x600), время с 200с по 600с

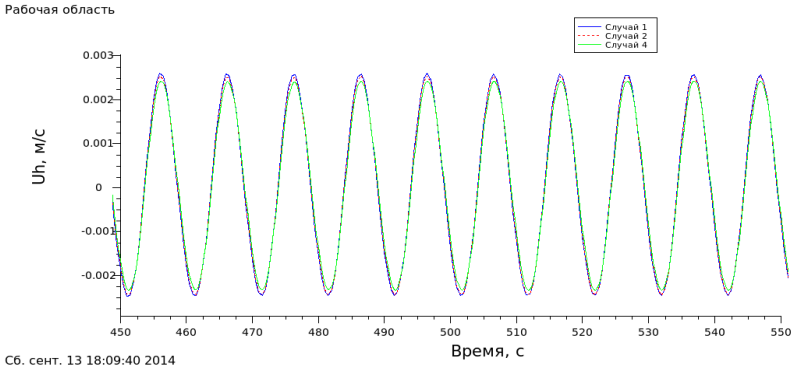


Рисунок 10. Сравнение динамики горизонтальной составляющей скорости среды в точке (28.2, 20) для трёх различных сеточных разрешений (225x150, 450x300, 900x600), время с 450с по 550с

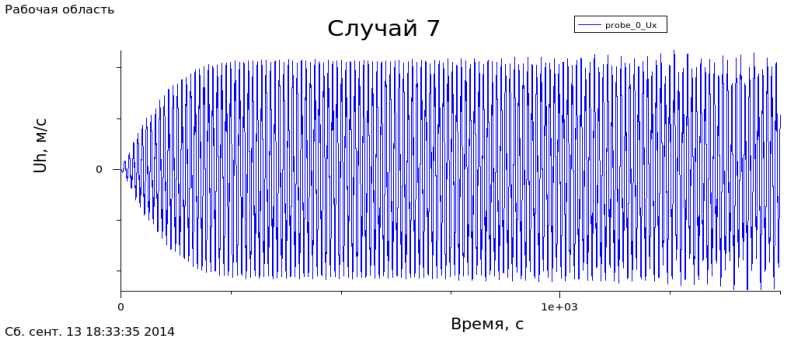


Рисунок 11. Динамика горизонтальной составляющей скорости в точке (28.2, 20) для случая 7 (без постепенного вывода скорости движения подвижной стенки)

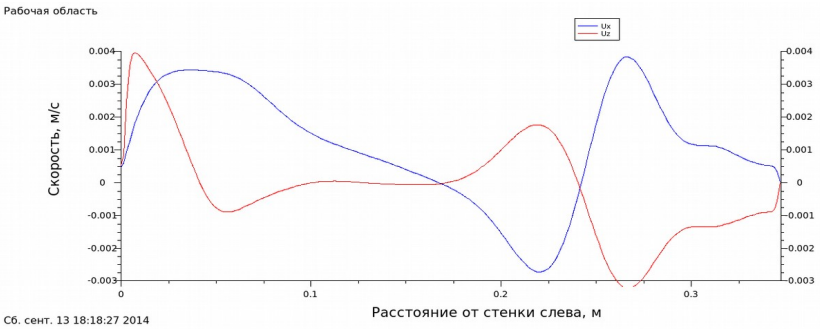


Рисунок 12. Профиль горизонтальной и вертикальной скоростей на линии y=20см для момента 501.5с (максимум горизонтальной скорости)

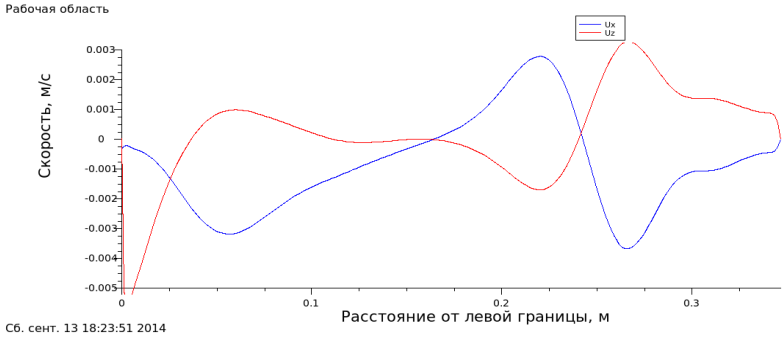


Рисунок 13. Профиль горизонтальной и вертикальной скоростей на линии y=20см для момента 506.5с (минимум горизонтальной скорости)

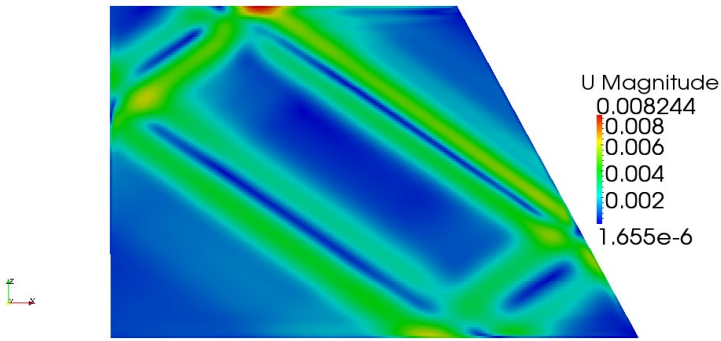


Рисунок 14. Визуализация мгновенного поля модуля скорости момента 501.5с (максимум горизонтальной скорости)

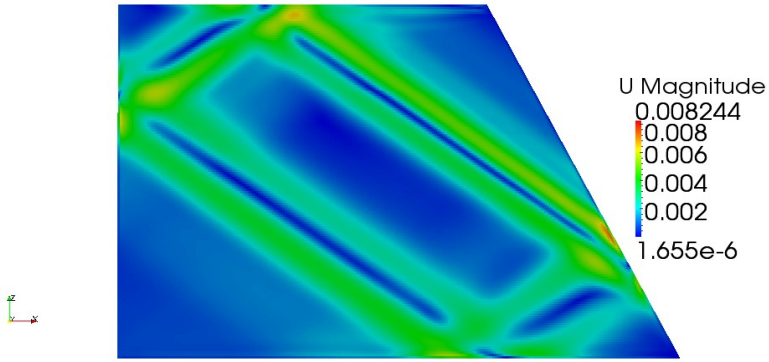


Рисунок 15. Визуализация мгновенного поля модуля скорости момента 506.5с (минимум горизонтальной скорости)

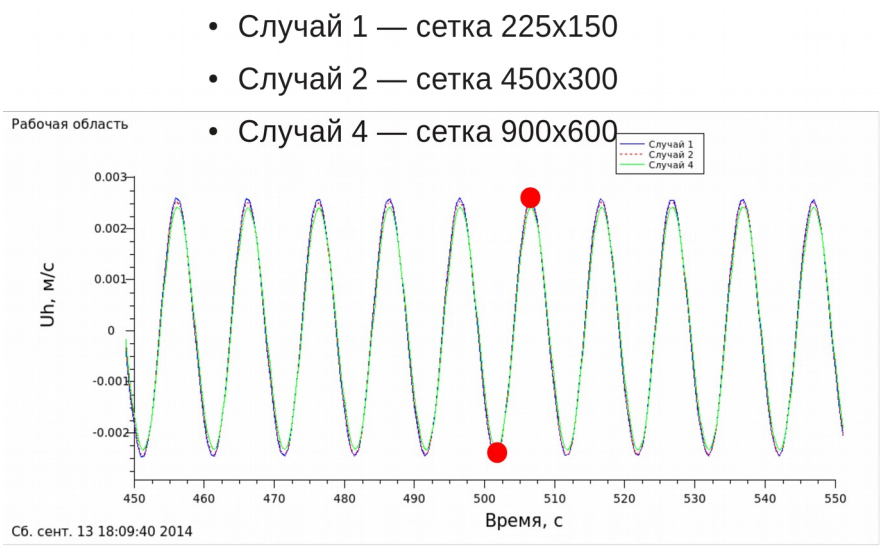


Рисунок 16. Сравнение сеточного разрешения и схема выбора точек для кросс-тестирования МКО и МСЭ

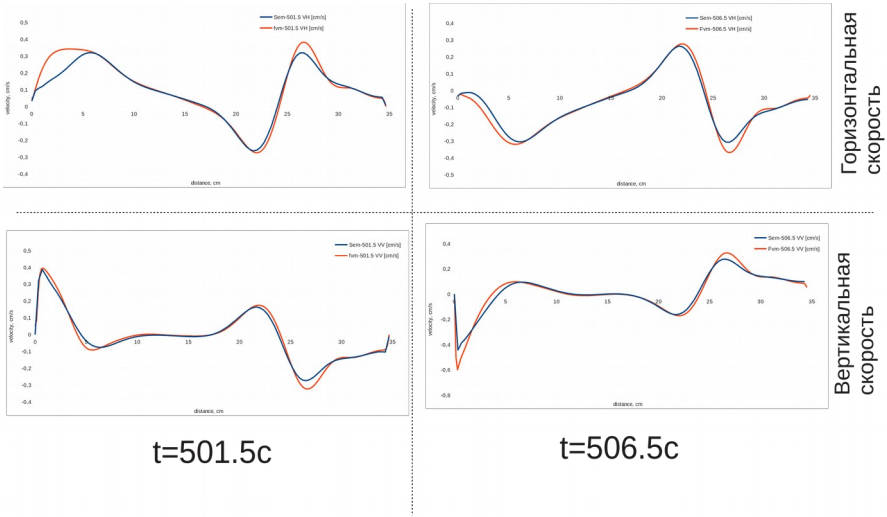


Рисунок 17. Сравнение МКО и МСЭ для различных моментов времени

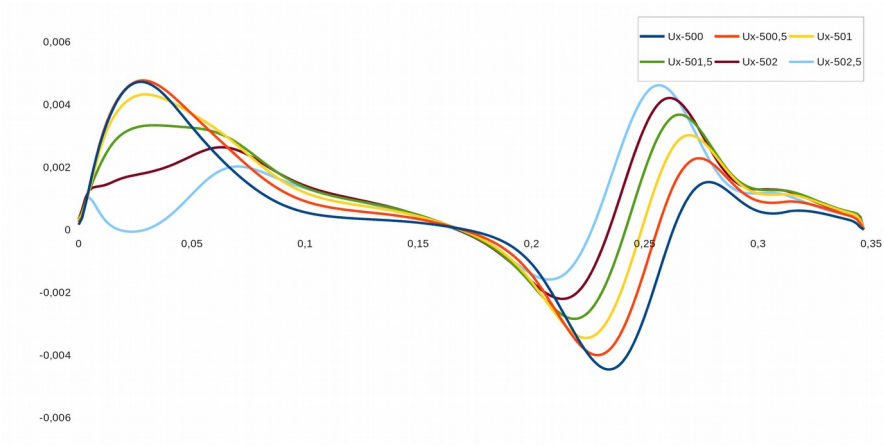


Рисунок 18. Изменение поля скорости продольной скорости среды в зависимости от времени

9. Выводы

Двумерные численные расчеты хорошо воспроизводят экспериментальные данные при учете изменения линейного профиля солености у свободной поверхности. При этом амплитуда колебаний в численных расчетах увеличена за счет потерь энергии волнопродуктора в трехмерном контейнере. При проведении эксперимента предполагалось создание линейного профиля, но добиться идеально линейной стратификации у верхней границы не представляется возможным. При учете изменения профиля солености у верхней границы форма аттрактора в расчете и эксперименте совпадает. Несмотря на то, что общий вид аттрактора хорошо воспроизводится как методом спектральных элементов, так и методом конечных объемов, в профилях скоростей имеются отличия у левой границы. Это требует дальнейшего исследования, поскольку такие отличия могут влиять на нелинейную динамику волн при развитии неустойчивостей.

В расчётах использовались вычислительные мощности платформы UniHUB ИСП РАН (http://www.unihub.ru) и суперкомпьютерных кластеров МГУ им. М.В,Ломоносова. Работы выполнялись при финансовой поддержке Министерства образования и науки Российской Федерации (уникальный идентификатор соглашения RFMEFI60714X0090) и гранта РФФИ 15-01-06363.

Список источников

[1]. Maas, L. R. M. & Lam, F.-P. A.// Geometric focusing of internal waves. J. Fluid Mech, 1995,. 300, 1–41

[2]. Dauxois, Thierry; Young, W.// Journal of Fluid Mechanics, 1999, vol. 390, Issue 01, p.271-295

[3]. Scolan, H., Ermanyuk, E., Dauxois, T.// 2013, Physical Review Letters, 110, 234501

[4]. Grisouard, N., Staquet, C., Pairaud, I.// 2008, Journal of Fluid Mechanics, 614, 1
[5]. Hazewinkel, J., van Breevoort, P., Dalziel, S.~B., Maas, L.~R.~M.// 2008, Journal of Fluid Mechanics, 598, 373
[6]. Frans-Peter A. Lam, Leo R.M. Maas. Internal wave focusing revisited; a reanalysis and new theoretical links // Fluid Dynamics Research 40 (2008) 95 – 122.
[7]. Mercier, Matthieu J.; Garnier, Nicolas B.; Dauxois, Thierry Reflection and diffraction of internal waves analyzed with the Hilbert transform Physics of Fluids, Volume 20, Issue 8, pp. 086601-086601-10 (2008).
[8]. Jouve, L., Ogilvie, G.~I.// Journal of Fluid Mechanics, 2014, 745, 223.
[9]. Fischer P., Ronquist E. Spectral element methods for large scale parallel Navier---Stokes calculations, Computer Methods in Applied Mechanics and Engineering, vol. 116, issue 1-4, pp. 69-76, 1994
[10]. Сибгатуллин И.Н. Моделирование волнового аттрактора в стратифицированной жидкости. Международная конференция "Mode Conversion, Coherent Structures and Turbulence", Space Research Institute of RAS, Moscow, Russia, 2014
[11]. Brouzet C., Dauxois T., Ermanyuk E., Kraposhin M., Sibgatullin I. Modelling of Wave Attractors in Stratified Fluids, 5-я международная научная школа молодых ученых «ВОЛНЫ И ВИХРИ В СЛОЖНЫХ СРЕДАХ», Москва, 2014
[12]. Fischer P.F. An overlapping schwarz method for spectral element solution of the incompressible Navier–Stokes equations. J. Comput. Phys. 133 (1), 84–101, 1997.
[13]. Загуменный Я.В., Чашечкин Ю.Д. Расчет течений непрерывно стратифицированной жидкости с использованием открытых вычислительных пакетов на базе технологической платформы UniHUB. Труды Института системного программирования РАН, том 24, 2013 г. стр. 87-106.

Direct numerical simulation of internal gravity wave attractor in trapezoidal domain with oscillating vertical wall

C. Brouzet [1] <christophe.brouzet@ens-lyon.fr>
T. Dauxois [1] < thierry.dauxois@ens-lyon.fr>
E. Ermanyuk [1,2] <ermanyuk@gmail.com>
S. Joubaud [1] <sylvain.joubaud@ens-lyon.fr>
M. Kraposhin [3,4] <os-cfd@yandex.ru>
I. Sibgatullin [4,5,6] <ilias_s@mail.ru>
[1] Laboratoire de Physique de l'École Normale Supérieure de Lyon, Universite de Lyon, France
[2] Lavrentyev Institute of Hydrodynamics, Novosibirsk, Russia
[3] P National Research Center "Kurchatow institute", Moscow, Russia
[4] Institute of system programming of Russian Academy of Sciences, Moscow, Russia
[5] Faculty of Mechanics and Mathematics, and Institute of Mechanics of Moscow State University, Russia
[6] Shirshov Oceanology Institute of Russian Academy of Sciences, Moscow, Russia

Abstract. Direct numerical simulation of internal gravity waves focusing and developement of a wave attractor was performed with the help of two different numerical approaches. Mathematical formulation corresponds to experiments on exitations of inner waves in a trapezoidal container with salt solutions through forced oscillations of the left boundary. It was shown that numerical simulations reproduce the experiments after taking into account ther imperfection of linear salinity profile near the upper boundary. The amplitudes of resulting oscillations in both numerical simulations are increased as compared to the experiments due to loss of energy of the 3D wave generator in the experiments. Despite the fact that the general shape of the attractor is reproduced by both method, there are differences in velocity profiles near the left boundary. This fact requires further investigations since this discrepancy may in influence nonlinear dynamics of developing instabilities.

Keywords: attractor, internal waves, gravity waves, DNS, direct numerical simulation.

References

[1]. Maas L. R. M. & Lam F.-P. A.// Geometric focusing of internal waves. J. Fluid Mech, 1995,. 300, 1–41
[14]. Dauxois Thierry, Young W.// Journal of Fluid Mechanics, 1999, vol. 390, Issue 01, p.271-295
[15]. Scolan H., Ermanyuk E., Dauxois T.// 2013, Physical Review Letters, 110, 234501

[16]. Grisouard N., Staquet C., Pairaud I.// 2008, Journal of Fluid Mechanics, 614, 1
[17]. Hazewinkel J., van Breevoort P., Dalziel S.B., Maas L.~R.~M.// 2008, Journal of Fluid Mechanics, 598, 373
[18]. Frans-Peter A. Lam, Leo R.M. Maas. Internal wave focusing revisited; a reanalysis and new theoretical links // Fluid Dynamics Research 40 (2008) 95 – 122.
[19]. Mercier Matthieu J., Garnier Nicolas B., Dauxois Thierry Reflection and diffraction of internal waves analyzed with the Hilbert transform Physics of Fluids, Volume 20, Issue 8, pp. 086601-086601-10 (2008).
[20]. Jouve L., Ogilvie G.I.// Journal of Fluid Mechanics, 2014, 745, 223.
[21]. Fischer P., Ronquist E. Spectral element methods for large scale parallel Navier---Stokes calculations, Computer Methods in Applied Mechanics and Engineering, vol. 116, issue 1-4, pp. 69-76, 1994
[22]. Sibgatullin I.N. Modeling of wave attractor in strafitied fluid. International conference "Mode Conversion, Coherent Structures and Turbulence", Space Research Institute of RAS, Moscow, Russia, 2014
[23]. Brouzet C., Dauxois T., Ermanyuk E., Kraposhin M., Sibgatullin I. Modelling of Wave Attractors in Stratified Fluids, 5-th international school of young scientists «Waves and vorticesin complex media», Moscow, 2014
[24]. Fischer P.F. An overlapping schwarz method for spectral element solution of the incompressible Navier–Stokes equations. J. Comput. Phys. 133 (1), 84–101, 1997.
[25]. Zagumennyi Ia.V. , Chashechkin Yu.D. Calculations of continuously stratified fluid flows using open source computational packages based on the technological platform UniHUB. Papers of Institute of System Programming of Russian Academy of Sciences, vol 24, 2013, pp. 87-106.

Исследование режимов виброкипящего гранулированного слоя с использованием пакета OpenFOAM

[1,2,3] *Н.С. Орлова <norlova.umi.vnc@gmail.com>*
[2] *Я.Н. Качалкина <yankevich-2502@mail.ru>*
[1] *Южный математический институт Владикавказского научного центра РАН и Правительства РСО-А, Россия, Владикавказ, ул. Маркуса, 22;*
[2] *Северо-Кавказский горно-металлургический институт (государственный технологический университет), Россия, Владикавказ, ул. Николаева, 44;*
[3] *Финансовый университет при Правительстве РФ, Россия, Владикавказ, ул. Молодежная, 7.*

Аннотация. Виброкипение широко используется в различных технологических процессах. В связи с этим исследование режимов виброкипения является актуальным. В работе проведено исследование режимов виброкипящего слоя с использованием пакета OpenFOAM. Представлены результаты моделирования динамики виброкипящего слоя частиц стекла при различных значениях амплитуды и частоты колебаний, толщины слоя засыпки. Выявлены режимы, при которых слой теряет устойчивость, и образуются всплески гранулированного материала и фонтанирующие каналы.

Ключевые слова: виброкипящий слой, гранулированный материал, математическое моделирование, OpenFOAM, twoPhaseEulerFoam, режимы виброкипения

1. Введение

В настоящее время имеющиеся модели виброкипения описывают поведение движения двухфазных сред только при определенных условиях [1-3], т.е. имеют различные области применения, которые зависят от входных параметров (толщина слоя засыпки материала, размер твердых частиц, амплитуда и частота колебаний полки, на которой располагается материал, и т.д.). Разработка универсальной трехмерной модели виброкипящего слоя имеет практическую значимость, так как за счет увеличения площади поверхности контакта фаз виброкипение широко используется в различных технологических процессах: очистка газов, химические технологии, сушка гранулированного материала. Несмотря на огромное практическое значение и

разнообразие работ в этом направлении, общепринятой универсальной модели пока нет.

В данной работе представлены результаты моделирования динамики виброкипящего слоя с использованием свободно распространяемого пакета для решения прикладных задач гидро и аэромеханики OpenFOAM при поддержке программы "Университетский кластер" с удаленным доступом к консоли на управляющем узле вычислительного кластера BL2×220 [4]. Для описания процесса виброкипения был доработан решатель twoPhaseEulerFoam, который использовался для моделирования динамики кипящего гранулированного слоя [5]. В решателе twoPhaseEulerFoam реализована двухжидкостная модель кипящего (ожиженного) слоя на основе континуального подхода (подхода Эйлера), при котором движение слоя рассматривается как движение двух взаимодействующих континуумов, связанных с газом и частицами. Основные уравнения двухжидкостной модели - уравнения неразрывности и уравнения количества движения для обеих фаз [5-8].

2. Постановка задачи

Для моделирования динамики виброкипящего слоя был доработан решатель twoPhaseEulerFoam, в котором вместо использования абсолютной системы отсчета для пространственных координат вводилась относительная система отсчета, движущаяся вместе с контейнером (полкой, на которой располагается слой материала). Предполагалось, что стенки контейнера не деформируются и перемещаются как твердое тело. В этой движущейся системе отсчета нет движения стенок. Таким образом, вычислительная сетка является статической в движущейся неинерциальной системе отсчета, что значительно упрощает процедуру численного решения уравнений. Концептуально, результаты должны быть эквивалентны результатам, полученным из решения основных уравнений в инерциальной системе координат с использованием сетки, которая движется со стенками. Поскольку рассматриваются вертикальные вибрации, считается, что будет меняться только вертикальная координата y. В связи с этим уравнения количества движения для обеих фаз примут следующий вид [9]:

$$\frac{\partial}{\partial t}\left(\alpha_\phi \overline{U}_\phi\right) + \nabla\cdot\left(\alpha_\phi \overline{U}_\phi \overline{U}_\phi\right) + \nabla\cdot\left(\alpha_\phi \overline{R}_{\phi,eff}\right) = -\frac{\alpha_\phi}{\rho_\phi}\nabla P + \alpha_\phi\left(\overline{g} - \overline{a}\right) + \overline{M}_\phi; \qquad (1)$$

где $\overline{a}$ - ускорение колебаний полки (стенки), на которой располагается слой материала. Т.к. рассматриваются только вертикальные колебания (вибрации), то ускорение колебаний определяется вертикальной составляющей

$\bar{a} = (a_x, a_y, a_z) = (0, a_y, 0)$, где $a_y = -A\omega^2 \sin(\omega t)$, $\omega = 2\pi f$ (A - амплитуда колебаний, f - частота колебаний).

Кроме того, решалось уравнение неразрывности (2) для обеих фаз.

$$\frac{\partial}{\partial t}(\alpha_\varphi) + \nabla \cdot (\alpha_\varphi \overline{U}_\varphi) = 0; \quad (2)$$

Индекс φ означает принадлежность к фазе (твердой «*a*» или газовой «*b*»); α_φ - объемная доля соответствующей фазы; ρ_φ - плотность фазы; $\overline{U}_\varphi$ - вектор скорости фазы; $\overline{R}_{\varphi,eff}$ - тензор эффективных напряжений; P - давление газовой фазы; $\overline{g}$ - ускорение свободного падения; $\overline{M}_\varphi$ - член, моделирующий обмен импульсом между фазами. Выражения для нахождения коэффициентов и членов, входящих в уравнения (1-2), подробно описаны в литературе [6-8].

В решателе twoPhaseEulerFoam обе фазы считаются несжимаемыми. Для газовой фазы реализована полуэмпирическая двухпараметрическая модель турбулентности k – ε [5,6]. Член, моделирующий обмен импульсом между фазами, определяется силами трения, возникающими между двумя фазами:

$$\overline{M}_\varphi = \frac{1}{\rho_\varphi} \overline{M}_{drag},$$

$$\overline{M}_{drag} = \alpha_a \alpha_b K \overline{U}_r,$$

где K – функция сопротивления, которая зависит от объемной доли газовой фазы. При $\alpha_b \le 0.8$ функция сопротивления определяется по формуле (3), а при $\alpha_b > 0.8$ - по формуле (4) [6-8].

$$K = 150 \frac{\alpha_a \mu_b}{(\alpha_b d_a)^2} + 1.75 \frac{\rho_b |\overline{U}_r|}{\alpha_b d_a}, \quad (3)$$

$$K = \frac{3}{4} C_D \frac{\rho_b \left| \overline{U}_r \right|}{d_a}, \tag{4}$$

где $\overline{U}_r = \overline{U}_a - \overline{U}_b$ - относительная скорость фаз, d_a – диаметр частиц, μ_b - динамическая вязкость газа, C_D - коэффициент сопротивления.

Для учета эффектов, обусловленных взаимодействием частиц друг с другом, используется кинетическая теория (по аналогии с кинетической теорией газа), с помощью которой можно выразить эффективные напряжения, возникающие в дисперсной фазе за счет движения частиц и за счет столкновений частиц друг с другом. По аналогии с термодинамической температурой, вводится гранулярная температура θ, как средняя энергия флуктуаций скорости частиц. В решателе для расчета гранулярной температуры используется алгебраическое уравнение (5) [7].

$$\theta = \left(\frac{-\left(K_1 \alpha_a + \rho_a\right) tr\left(\overline{\overline{D}}_s\right) + \sqrt{\left(K_1 \alpha_a + \rho_a\right)^2 tr^2\left(\overline{\overline{D}}_s\right) + 4K_4 \alpha_a \left[2K_3 tr\left(\overline{\overline{D}}_s^2\right) + K_2 tr^2\left(\overline{\overline{D}}_s\right)\right]}}{2\alpha_a K_4} \right)^2, \tag{5}$$

В уравнении (5) $\overline{\overline{D}}_s$ - тензор скоростей деформации твердой фазы, коэффициенты $K_1 - K_4$ определяются следующим образом:

$$\overline{\overline{D}}_s = \frac{1}{2}\left[\nabla \overline{U}_a + \left(\nabla \overline{U}_a\right)^{\mathrm{T}}\right],$$

$$K_1 = 2(1+e)\rho_a g_0,$$

$$K_2 = \frac{4}{3\sqrt{\pi}} d_a \rho_a (1+e)\alpha_a g_0 - \frac{2}{3} K_3,$$

$$K_3 = \frac{d_a \rho_a}{2}\left(\frac{\sqrt{\pi}}{3(3-e)}\left[1+\frac{2}{5}(1+e)(3e-1)\alpha_a g_0\right]+\frac{8\alpha_a}{5\sqrt{\pi}} g_0(1+e)\right),$$

$$K_4 = \frac{12(1-e)^2 \rho_a g_0}{d_a\sqrt{\pi}},$$

$$g_0 = \frac{3}{5}\left[1-\left(\frac{\alpha_a}{\alpha_{a\max}}\right)^{\frac{1}{3}}\right]^{-1},$$

где g_0 - радиальная функция контакта, $\alpha_{a\max}$ - максимальное значение объемной доли частиц, e - коэффициент восстановления в случае столкновений частица-частица. Более подробное описание коэффициентов представлено в работах [7,8].

3. Начальные и граничные условия

Задача решалась в трехмерном приближении. В связи с тем, что введена относительная система отсчета, используются следующие начальные условия:

$$\overline{U}_a = 0;\ \overline{U}_b = 0;\ P = 0;\ \alpha_a = 0{,}6;\ \theta = 0.$$

Далее представлены граничные условия на левой, правой, передней и задней стенках, а также на нижней стенке (полке):

$$\overline{U}_a = 0;\ \overline{U}_b = 0;\ \nabla P = 0;\ \nabla\alpha_a = 0;\ \nabla\theta = 0.$$

Граничные условия на верхней свободной поверхности:

$$\nabla\overline{U}_a = 0;\ \nabla\overline{U}_b = 0;\ \nabla P = 0;\ \nabla\alpha_a = 0;\ \nabla\theta = 0.$$

Размеры вычислительной области: высота – 0,4 м, ширина - 0.4 м, толщина - 0.4 м. Процесс виброкипения рассчитывался за 2 с. При этом использовался шаг по времени, равный 1×10^{-4} с. Шаг по координате x равен 0.005 м, шаг по координате y - 0.005 м, шаг по координате z равен 0.005 м. Проводилось распараллеливание вычислений на 12 ядрах. В таблице 1. представлены

значения входных параметров задачи. В качестве твердой фазы были выбраны сферические частицы стекла.

Табл. 1. – Значения входных параметров задачи.

	Описание	Значение
1	плотность твердой фазы (частицы стекла), ρ_a [кг/м^3]	2600
2	плотность газовой фазы (воздуха), ρ_b [кг/м^3]	1,2
3	вязкость газовой фазы (воздуха), μ_b [Па×с]	$1{,}5\times10^{-5}$
4	коэффициент восстановления частица-частица, e [-]	0,9
5	начальное значение объемной доли частиц в слое, α_{a0} [-]	0,6
6	начальная высота слоя, H_0 [м]	0,05-0,1
7	ширина слоя, L[м]	0,4
8	толщина слоя, W[м]	0,4
9	диаметр частиц, d_a[м]	0,0003

4. Результаты расчетов

С использованием разработанной трехмерной модели виброкипящего слоя было проведено исследование режимов виброкипения. Расчеты проводились при значениях амплитуды колебаний в диапазоне A = 1,5 – 9 мм, частоты колебаний в диапазоне f = 20 – 60 Гц при значениях начальной толщины слоя частиц 50 – 100 мм. В результате серии трехмерных вычислительных экспериментов с распараллеливанием на супер-ЭВМ наблюдается волнообразная поверхность гранулированного материала. С увеличением амплитуды и частоты колебаний слой частиц теряет устойчивость и образуются всплески гранулированного материала.

На рис. 1 представлены результаты моделирования динамики виброкипящего слоя в момент времени 1,5 с при различных значениях амплитуды колебаний A и при частоте колебаний f = 20 Гц для слоя с толщиной засыпки 50 мм.

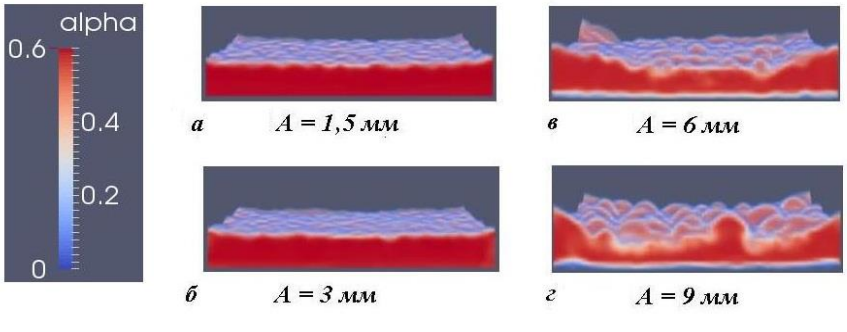


Рис. 1. Структура виброкипящего слоя при разных значениях амплитуды колебаний.

Из рис. 1 видно, что с увеличением амплитуды колебаний A > 3мм начинают появляться всплески гранулированного материала, а при амплитуде A = 9 мм всплески больше похожи на фонтанирующие каналы.

На рис. 2 представлены аналогичные результаты, полученные при частоте колебаний f = 40 Гц.

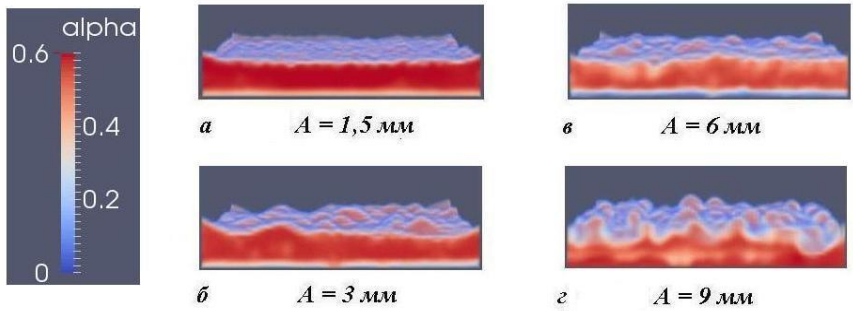


Рис. 2. Структура виброкипящего слоя при разных значениях амплитуды колебаний.

В отличие от предыдущего случая (рис. 1) всплески гранулированного материала появляются уже при амплитуде колебаний A = 3 мм. А при амплитуде A = 9 мм наблюдаются явно выраженные фонтанирующие каналы.

Очевидно, что образование всплесков в большей степени зависит от амплитуды колебаний, чем от частоты. Они начинают появляться при значениях амплитуды A > 2 мм. При значениях амплитуды колебаний A > 6 мм в слое наблюдаются локализованные фонтанирующие каналы.

С дальнейшим увеличением частоты колебаний (до 60 Гц) при высоких значениях амплитуды (> 6 мм) количество фонтанирующих каналов

возрастает, зазор между нижней частью слоя и полкой (на которой располагается материал) становится больше.

Вычислительные эксперименты проводились и для более толстых слоев (с толщиной засыпки до 100 мм). На рис. 3 представлены результаты сравнения степени расширения виброкипящего слоя с толщиной засыпки 50 мм (кривая 1) и 100 мм (кривая 2) при частоте колебаний 20 Гц и при амплитуде колебаний 1,5 - 9 мм.

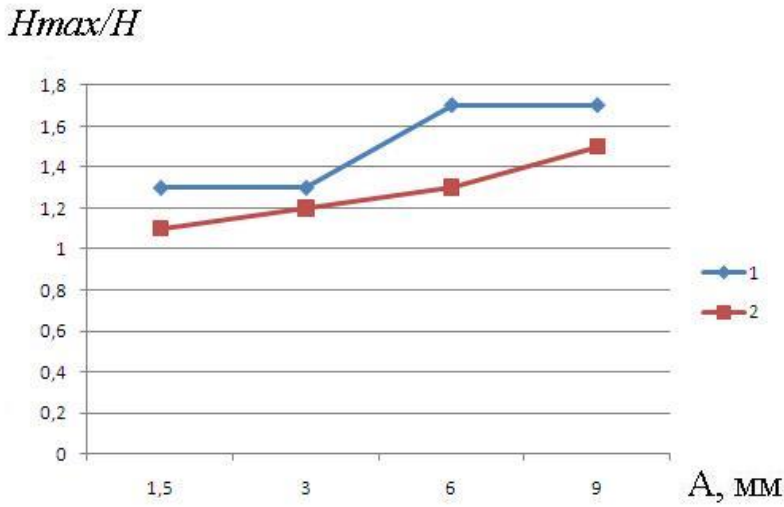


Рис. 3. Степень расширения виброкипящего слоя

Качественно результаты расчетов, полученные при различных значениях амплитуды и частоты колебаний для слоя с толщиной засыпки 100 мм мало отличаются от результатов, полученных для слоя с толщиной 50 мм. С увеличением амплитуды и частоты колебаний в слое наблюдаются всплески (при $A > 2$ мм) и фонтанирующие каналы (при $A > 6$ мм). Но степень расширения более толстого слоя (100 мм) меньше, чем степень расширения относительно тонкого (50 мм), как видно из рис. 3.

5. Заключение

В результате серии вычислительных экспериментов наблюдается волнообразная поверхность гранулированного материала. С увеличением амплитуды и частоты колебаний слой частиц теряет устойчивость и образуются всплески гранулированного материала. Образование всплесков в большей степени зависит от амплитуды колебаний, чем от частоты. Они начинают появляться при значениях амплитуды A > 2 мм. При значениях

амплитуды колебаний A > 6 мм в слое наблюдаются локализованные фонтанирующие каналы.

Таким образом, в результате моделирования виброкипящего слоя с использованием пакета OpenFOAM исследованы режимы виброкипения. Выявлены режимы, при которых слой теряет устойчивость и образуются всплески гранулированного материала и фонтанирующие каналы.

Список литературы

[1]. Н.С. Орлова. Тестирование двух моделей виброожиженного слоя. Известия вузов. Северо-Кавказский регион. Технические науки. №2, 2012 г. стр.42-45.

[2]. Н.С. Орлова. Сравнение расчетов по двухжидкостной модели виброожиженного слоя с экспериментальными данными. Инженерно - физический журнал. том 85, №6, 2012 г. стр.1202-1207.

[3]. Н.С. Орлова. Сравнение результатов экспериментального исследования виброкипящего слоя с расчетами по гидродинамической модели гранулярного газа. Инженерно- физический журнал. Т. 87, № 2, 2014. стр. 429-435.

[4]. Консоль на управляющий узел кластера (2012), "BL2x220 Cluster Console" // URL: https://unihub.ru/resources/bl2x220cc (дата обращения: 10.06.2014).

[5]. Е.С. Каменецкий, Н.С. Орлова, М.В. Волик, Д.Г. Минасян. Исследование динамики кипящего гранулированного слоя с использованием пакета OpenFoam. Известия вузов. Северо-Кавказский регион. Естественные науки. №5, 2014 г. стр. 37 – 42.

[6]. H. Rusche. Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions. Thesis submitted for the degree of Doctor of Philosophy of the University of London and Diploma of Imperial College. 2002. 343 p.

[7]. B. van Wachem. Derivation, Implementation, and Validation of Computer Simulation Models for Gas-Solid Fluidized Beds – Dissertation at Delft University of Technology. 2000. 222 p.

[8]. D. Gidaspow. Multiphase flow and fluidization: Continuum and kinetic theory descriptions. Boston: Academic Press Inc., 1994. 211 p.

[9]. N.S. Orlova, M.V. Volik. Modelling of vibrofluidized bed dynamics using OpenFoam. Waves and vortices in complex media: 5-th International Scientific School of Young Scientists, November 25-28, 2014, Moscow. M.: MAKS Press, 2014. pp. 72-74.

Investigation of vibrofluidized granular layer modes using OpenFOAM

[1,2,3] *N.S. Orlova <norlova.umi.vnc@gmail.com>*
[2] *Ya.N. Kachalkina <yankevich-2502@mail.ru>*
[1] *South Mathematical Institute of the Vladikavkaz Scientific Center of the Russian Academy of Sciences and the Government of Republic of North Ossetia-Alania, Russia, Vladikavkaz, Marcus street, 22*
[2] *North-Caucasian Mining and Metallurgical Institute (State Technological University), Russia, Vladikavkaz, Nikolaev street, 44*
[3] *Financial University under the Government of the Russian Federation, Russia, Vladikavkaz, Molodezhnaya street, 7*

Abstract. Vibrofluidization is widely used in various industrial processes. So the study of vibrofluidization modes is current research. In this paper we investigated the modes of vibrofluidized layer using the package OpenFOAM. The results of modelling of the vibrofluidized layer dynamics at different values of the amplitude and frequency of oscillations, the thickness of the filling, were presented. The modes in which layer becomes unstable were identified. Wavelets formed from granular material and the jetting channels appear in these modes.

Key words: vibrofluidized layer, granular material, mathematical modeling, OpenFOAM, twoPhaseEulerFoam, vibrofluidization modes

References

[1]. N.S. Orlova. Testirovanie dvuh modeley vibroozhizhennogo sloya. [Testing of two models of vibrofluidized layer]. Izvestiya vuzov. Severo-Kavkazskiy region. Tehnicheskie nauki. [Proceedings of the universities. North Caucasus region. technical sciences] 2012, no. 2, pp. 42-45 (in Russian)
[2]. N. S. Orlova. Comparison of calculations by the two-field vibrofluidized-bed model with experimental data. Journal of Engineering Physics and Thermophysics vol. 85, no. 6, pp 1305-1310. doi 10.1007/s10891-012-0775-x
[3]. N. S. Orlova. Comparison of the Results of Experimental Investigations of a Vibrofluidized Bed with Calculations by a Granular Gas Hydrodynamic Model. Journal of Engineering Physics and Thermophysics vol. 87, no. 2, pp 443-449. doi 10.1007/s10891-014-1030-4
[4]. Konsol na upravlyayuschiy uzel klastera (2012), "BL2x220 Cluster Console" // URL: https://unihub.ru/resources/bl2x220cc (data obrascheniya: 10.06.2014). [Console to the

control node of the cluster (2012), "BL2x220 Cluster Console" // URL: https://unihub.ru/resources/bl2x220cc (date of treatment: 10.06.2014).]

[5]. E.S. Kamenetskiy, N.S. Orlova, M.V. Volik, D.G. Minasyan. Issledovanie dinamiki kipyaschego granulirovannogo sloya s ispolzovaniem paketa OpenFoam. [The study of the dynamics of fluidized granular layer using the package OpenFoam.] Izvestiya vuzov. Severo-Kavkazskiy region. Estestvennyie nauki. [Proceedings of the universities. North Caucasus region. natural sciences] 2014, no. 5, pp. 37-42 (in Russian)

[6]. H. Rusche. Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions. Thesis submitted for the degree of Doctor of Philosophy of the University of London and Diploma of Imperial College. 2002. 343 p.

[7]. B. van Wachem. Derivation, Implementation, and Validation of Computer Simulation Models for Gas-Solid Fluidized Beds – Dissertation at Delft University of Technology. 2000. 222 p.

[8]. D. Gidaspow. Multiphase flow and fluidization: Continuum and kinetic theory descriptions. Boston: Academic Press Inc., 1994. 211 p.

[9]. N.S. Orlova, M.V. Volik. Modelling of vibrofluidized bed dynamics using OpenFoam. Waves and vortices in complex media: 5-th International Scientific School of Young Scientists, November 25-28, 2014, Moscow. M.: MAKS Press, 2014. pp. 72-74.

Применение графических ускорителей для расчета гидродинамических характеристик гребных винтов в пакете OpenFOAM

[1,2] *Б.И. Краснопольский <krasnopolsky@imec.msu.ru>*
[2] *А.В. Медведев <alexey.v.medvedev@gmail.com>*
[1] *А.Ю. Чулюнин <chulyu-n@mail.ru>*
[1] *Институт механики МГУ имени М.В. Ломоносова,*
119192, Россия, г. Москва, Мичуринский пр-т, д. 1.
[2] *ЗАО «Т-Сервисы»,*
117198, Россия, г. Москва, Ленинский пр-т, д. 113/1.

Аннотация. Пакет OpenFOAM является одним из популярных инженерных инструментов для численного моделирования задач прикладной гидродинамики, для которых могут быть характерны сложные геометрии и сетки с числом ячеек, измеряемых десятками миллионов. Поскольку решение такого рода задач зачастую отличается большой продолжительностью и ресурсоёмкостью, любое ускорение таких расчетов имеет большое практическое значение. На основе одной практической задаче численного моделирования гидродинамических характеристик гребных винтов в настоящей работе исследуется вопрос оптимизации расчета в OpenFOAM за счет применения оригинальной библиотеки SparseLinSol (SLS), разрабатываемой авторами. Библиотека предназначена для решения больших разреженных систем уравнений на суперкомпьютерах и использует итерационные методы подпространства Крылова и многосеточные методы. Алгоритмы библиотеки используют оригинальную гибридную схему распараллеливания, комбинирующию модели MPI и Posix Shared Memory, а также допускают использование графических ускорителей NVIDIA для значительной части реализованных методов. В результате проведенного тестирования на вычислительной системе, оборудованной ускорителями NVIDIA X2070, показано, что: 1) результаты моделирования целевой задачи в пакете OpenFOAM, в целом, соответствуют результатам, полученным в пакете Star-CCM и результатам экспериментов; 2) реализованные методы решения СЛАУ обладают большей робастностью по сравнению с многосеточным методом GAMG, реализованным в пакете OpenFOAM; 3) Гибрдная модель распараллеливания значительно улучшает масштабируемость солвера, что позволяет добиваться линейной масштабируемости до 128 узлов, на всем диапазоне рассмотренном в проведенных тестах; 4) использование графических ускорителей способно увеличить скорость расчётов в 1.4-3 раза; 5) реализация методов в библиотеке SparseLinSol превосходит по скорости реализацию методов из библиотеки hypre для той же комбинации методов и тесторых матриц.

Ключевые слова: многосеточные методы; графические ускорители; гребные винты; масштабируемость; пакет OpenFOAM.

1. Введение

Решение прикладных инженерных и научных задач вычислительной гидродинамики в настоящее время является одной из типичных задач для многопроцессорных вычислительных систем. Научным расчетам свойственны достаточно простые геометрии расчетных областей, но более “аккуратные” численные процедуры, как то схемы высоких порядков аппроксимации по времени и пространству, быстрые методы решения систем уравнений, оптимизированные под конкретную геометрию расчетной области, более достоверные модели турбулентности и прочее. Для проведения таких расчетов зачастую разрабатываются авторские «in-house» расчетные алгоритмы и коды, ориентированные на изучение каких-то конкретных физических явлений. Инженерные расчеты отличает комплексная геометрия объектов и сложные конфигурации расчетных областей, что накладывает определенные ограничения на выбор численных методов. Приоритетным здесь является использование методов, обладающих достаточной универсальностью и гибкостью, чтобы обеспечить решение поставленной задачи даже для сложных расчетных областей. Такой функционал и требования к методам обычно отличаются от традиционных «in-house» кодов, и реализован в инженерных CFD-пакетах.

Одним из примеров расчетной области со сложной конфигурацией является задача о моделировании гидродинамических характеристик гребных винтов судна [1, 2]. Расчет гидродинамических характеристик пары гребных винтов предполагает проведение нестационарного расчета с вращением подобластей сетки, ассоциированных с движущимися винтами. В качестве инструментария для численного моделирования в работе использовался пакет с открытым исходным кодом OpenFOAM. Для корректного осреднения результатов расчетов необходимо смоделировать как минимум несколько оборотов винтов, что приводит к расчету десятков и сотен тысяч шагов по времени. С учетом того, что для удовлетворительного разрешения подобных расчетных областей требуются сетки порядка 10^7-10^8 ячеек, а расчет одного шага по времени занимает несколько минут, то крайне актуальным оказывается вопрос ускорения подобных расчетов. Проведенные оценки показали, что до 90% расчета шага по времени может занимать решение систем линейных алгебраических уравнений (СЛАУ) для поправки давления. Данное наблюдение очевидным образом указывает на наиболее перспективную область исследований с целью ускорения расчетов. Этап расчета, связанный с решением СЛАУ, в достаточной степени обособлен в коде пакета OpenFOAM, что позволяет применять пользовательские динамически подключаемые библиотеки численных методов без внесения изменений в исходный код пакета.

Потенциальное ускорение этапа решения СЛАУ для поправки давления, по сравнению со стандартными методами из пакета OpenFOAM, а именно методами GAMG и/или итерационными методами подпространства Крылова, может быть достигнуто за счет нескольких факторов. К их числу можно отнести как использование более эффективных математических методов, так и попытки применения более эффективных моделей программирования для улучшения масштабируемости методов, а также использования сопроцессоров или ускорителей. В данной работе будет затронут каждый из указанных аспектов на примере разрабатываемой авторами библиотеки численных методов для решения больших разреженных систем линейных алгебраических уравнений SparseLinSol.

Говоря об использовании графических ускорителей, следует изначально корректно оценивать потенциал ускорения для задач линейной алгебры с разреженными матрицами, не ожидая 20-40-кратных ускорений. Поскольку основными операциями в этих алгоритмах являются операции типа умножения матрицы на вектор, то данные задачи относятся к классу «memory bound» приложений, и основные вопросы ускорения вычислений будут связаны с организацией доступа к памяти и пропускной способностью шины памяти сопроцессоров. Так, для графических ускорителей NVIDIA Tesla поколения Fermi пропускная способность составляет до 178 ГБ/сек в режиме ECC OFF (порядка 150 ГБ/сек для более применимого на практике режима с коррекцией ошибок), и порядка 250 ГБ/сек для более современного поколения Kepler. По сравнению с типичным значением для центральных процессоров порядка 50 ГБ/сек, потенциальное ускорение от использования одного сопроцессора, с учетом различных накладных расходов, следует ожидать в пределах 2.5-4 раз по сравнению с одним центральным процессором.

2. Постановка задачи

В настоящей работе исследовалась задача об определении интегральных характеристик гребных винтов в постановке в открытой воде: обтекание набегающим потоком рулевой колонки с парой разнонаправленно вращающихся винтов. В рамках построенной модели для проведения численного моделирования выделяется расчетная область в виде усеченного конуса с высотой H=0.705 м. и диаметрами оснований d_1=0.367 м. и d_2=0.4 м. соответственно. Для расчета нестационарного течения вязкой несжимаемой жидкости (вода при стандартных условиях) использовались осредненные нестационарные уравнения Рейнольдса, замкнутые с помощью модели турбулентности SST [3]. На основании конуса с диаметром d_1 задается нормальная компонента скорости U=3 м/с, а на противоположном основании d_2 фиксируется распределение давления P=*const*. На твердых стенках внутри расчетной области задается условие «прилипания», а на боковой поверхности конуса предполагается условие непротекания и отсутствия касательных напряжений («симметрия»). Сопряжение вращающихся подобластей сетки

реализовано с использованием технологии AMI-патчей и соответствующего решателя pimpleDyMFoam.

В ходе работы были построены три расчетные сетки общим размером 41, 60 и 99 млн. ячеек. Данные сетки отличаются степенью сгущения ячеек у поверхности лопастей винтов и предназначены для исследования сеточной сходимости результатов расчетов. При этом, высота пристеночных ячеек для указанных сеток составила 3 мкм (рис. 2).

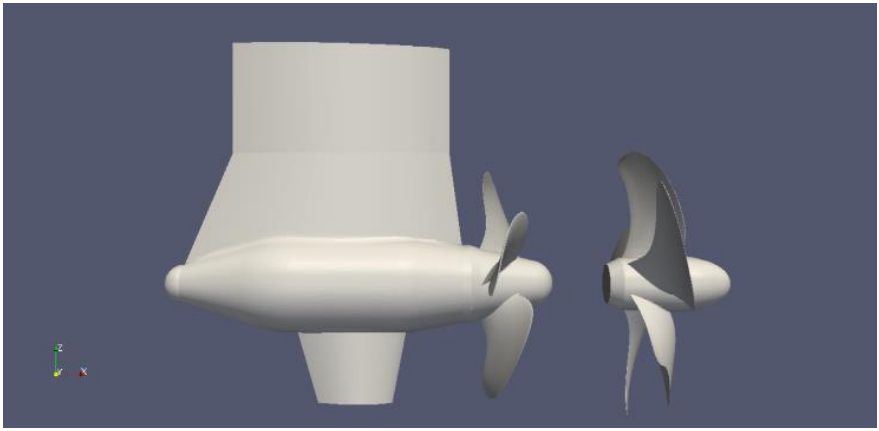


Рис. 1. Сборка модели.

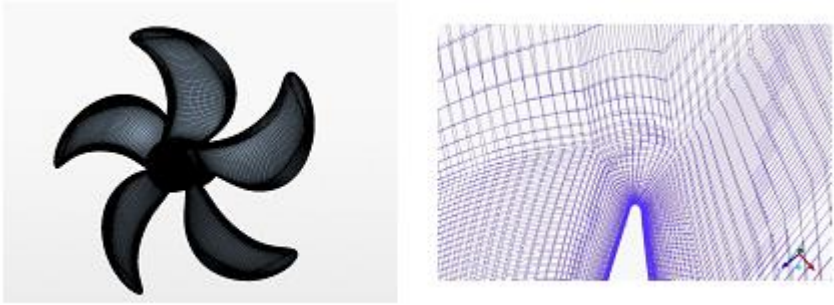


Рис. 2. Расчетная сетка на поверхности гребного винта и фрагмент расчетной сетки в окрестности кромки лопасти винта.

Расчеты проводились в пакете OpenFOAM версии 2.2.1, установленном на вычислительной системе «Ломоносов» суперкомпьютерного центра МГУ (табл. 1). Процедура расчета шага по времени требовала определенного количества внутренних итераций, причем матрица системы уравнений для поправки давления менялась не на каждой такой итерации. Суммарно, на шаге

по времени выполнялось решение 20 систем уравнений с 5 различными правыми частями для каждой из них.

Для стандартных методов решения систем уравнений, реализованных в пакете OpenFOAM, рассматривались две конфигурации с многосеточным методом в качестве решателя (GAMG) и методом сопряженных градиентов с многосеточным методом в качестве предобуславливателя (PCG+GAMG). В обоих случаях использовался следующий набор параметров многосеточного метода:

```
pressure_solver
{
 solver                  GAMG;
 smoother                DIC;

 cacheAgglomeration      on;
 nPreSweeps              0;
 nPostSweeps             2;
 nCellsInCoarsestLevel   1000;
 agglomerator            faceAreaPair;
 mergeLevels             1;
}
```

Точность решения систем уравнений для давления по норме, принятой в пакете OpenFOAM, ограничивалась значением $\|\mathbf{r_n}\|_{OF} < 10^{-7}$. Отдельные эксперименты, проведенные для комбинации метода сопряженных градиентов с частичным разложением Холецкого, показали, что данные методы не всегда обеспечивают удовлетворительную скорость сходимости к решению.

Табл. 1. Основные параметры тестовой вычислительной платформы.

Вычислительная система	«Ломоносов»
Процессор	2xX5570, 8 ядер/узел
Графический ускоритель	2xNVIDIA Tesla X2070
Оперативная память	8 ГБ
Interconnect	Infiniband QDR
Компилятор	Intel compilers, 13.1.0
Библиотека MPI	Intel MPI, 4.1.0

3. Библиотека SparseLinSol

В качестве внешней библиотеки в работе была использована разрабатываемая авторами библиотека *SparseLinSol (SLS)*. Данная библиотека содержит параллельные реализации итерационных методов подпространства Крылова и

многосеточных методов. Для центральных процессоров разработаны гибридные реализации методов, основанные на модели программирования MPI+Posix Shared Memory. Также ведется работа по переносу части вычислений и использованию в расчетах графических ускорителей.

В библиотеке SLS в данных расчетах использовалась конфигурация стабилизированного метода бисопряженных градиентов (BiCGStab) [4] с классическим алгебраическим многосеточным предобуславливателем (CAMG) [5]. Для построения иерархии матриц многосеточного метода использовался ряд алгоритмов из библиотеки hypre [6].

Для использования библиотеки в расчетах в пакете OpenFOAM был разработан соответствующий плагин сопряжения. В ходе его реализации было заложено условие, чтобы для его использования не требовалось внесения каких-либо изменений в исходный код и перекомпиляции всего кода пакета OpenFOAM. В данном плагине предусмотрена поддержка *processor*- и *cyclicAMI*-патчей, что позволяет использовать библиотеку *SparseLinSol* для расчетов задач с AMI-интерфейсами в распределенном режиме.

3.1 Особенности реализации методов на графических ускорителях

Среди основных трудностей при реализации типичных для задач линейной алгебры с разреженными матрицами алгоритмов на графических ускорителях отметим следующие:

1) выбор формата представления разреженных матриц;
2) накладные расходы, связанные с работой на графических ускорителях;
3) эффективная реализация распределенных вариантов алгоритмов.

3.1.1 Выбор формата представления матриц

Из множества известных вариантов форматов представления матриц для операций разреженной линейной алгебры наиболее универсальным можно считать формат CSR, обеспечивающий приемлемые результаты производительности операций на матрицах практически любого типа. Проблема, однако, заключается в том, что производительность операций с матрицами при использовании этого формата на графических ускорителях не достигает максимальной производительности. Причина обусловлена особенностями работы подсистемы памяти графических ускорителей, которая требует соблюдения правила «coalescing» для операций чтения и записи чтобы пропускная способность памяти была использована полностью. Известные алгоритмы, например, операции умножения матрицы на вектор (SpMV) для формата CSR, такие как алгоритм Н. Белла [7], не могут обеспечить выполнения правила «coalescing» для всех видов матриц, поэтому в худших случаях эффективность использования пропускной способности памяти для

них может снижаться относительно теоретического максимума в несколько раз.

Среди множества разработанных форматов представления разреженных матриц, допускающих более эффективные реализации алгоритмов типа SpMV, можно выделить семейство ELL-форматов. Хотя оригинальный формат ELLPACK имеет достаточно ограниченную область применимости для матриц произвольного вида, на его основе разработан ряд модификаций и обобщений, например [8], делающие этот формат более применимым на практике. Одним из альтернативных подходов, который должен обеспечить приемлемую производительность на графических ускорителях для любого типа разреженных матриц, представляется комбинирование форматов типа ELLPACK и CSR – представление матрицы в виде суперпозиции нескольких форматов, так чтобы наиболее подходящие алгоритмы работали для подходящих частей матриц. Недостатком такой методики становится тот факт, что необходимо реализовывать не один, а серию алгоритмов для каждой операции вместе с алгоритмами выбора типа представления в зависимости от параметров матрицы, и несколько алгоритмов конвертации форматов. Тем не менее, такой подход представляется на данный момент наиболее приемлемым компромиссом.

Библиотека SparseLinSol содержит реализации базовых алгоритмов для форматов CSR и ELLPACK. Эксперименты с реальными задачами, в том числе обсуждаемыми в настоящей работе, показывают, что для более чем половины матриц, операции с которыми выполняются в ходе выполнения многосеточного метода, формат ELLPACK является достаточно эффективным и именно ELLPACK-представление предпочтительнее для реализации методов на графических ускорителях. Для оставшейся части матриц CSR формат показывает лучшие результаты. Алгоритмы совмещения разных форматов представления матриц в ходе расчета на данный момент находятся в стадии разработки, и все имеющиеся результаты получены на версии библиотеки, использующей ELLPACK представление матриц на графических ускорителях.

3.1.2 Проблема накладных расходов, связанных с работой на графических ускорителях

Поскольку графический ускоритель является другой архитектурой по отношению к центральным процессорам, а также интегрирован в вычислительную систему фактически как периферийное устройство, находящееся на интерфейсной шине общего назначения, взаимодействие с ним не лишено некоторых накладных расходов. В ходе разработки и отладки алгоритмов выяснилось [2], что есть виды таких расходов, которые трудно или невозможно устранить, и они делятся на две категории: 1) временн*а*я задержка, связанная с запуском вычислительного ядра на графическом ускорителе и 2) временн*а*я задержка, связанная с копированием данных из

оперативной памяти вычислительного узла в память ускорителя или обратно. Запуск «пустого» вычислительного ядра на системах, оснащённых адаптерами NVIDIA, отнимает не менее 2-3 мкс, однако эта цифра может меняться и в большую сторону при увеличении числа нитей. Однако, есть основания полагать, что накладные расходы на запуск реальных ядер могут быть существенно выше в зависимости от числа используемых ядром регистров или разделяемой памяти. К сожалению, документация системного программного обеспечения графических ускорителей не позволяет оценить эти цифры точнее. Накладные расходы на выполнение описанных выше операций копирования варьируются слабо и составляют, по нашим оценкам, величины порядка 5 мкс (не считая собственно времени копирования, которое определяется пропускной способностью шины PCI-E). Сложности, связанные с этими накладными расходами состоят в том, что от них трудно или невозможно избавиться на практике, и они в итоге, в соответствии с законом Амдала, сказываются на масштабируемости распределенных вариантов разрабатываемых алгоритмов.

3.1.3 Распределенная реализация методов для графических ускорителей

Масштабируемая реализация распределенных вариантов алгоритмов разреженной линейной алгебры, таких как умножение матрицы на вектор, требует детальной проработки, как минимум, двух следующих вопросов: 1) как сделать, чтобы все коммуникации между узлами перекрывались по времени с вычислительными процедурами, так чтобы время на выполнение коммуникаций не вносило вклад в общее время выполнения алгоритма; 2) как уменьшить объём и количество коммуникаций между узлами.

Для того, чтобы реализовать идею перекрытия коммуникаций и вычислений, в библиотеке SparseLinSol реализована дополнительная процедура предварительной сегментации матрицы на блоки по столбцам. Принципы сегментации сходны с описанными в [9]. Вычисления для каждого сегмента могут таким образом быть осуществлены по мере получения необходимой для этой операции части вектора от соседей, а сама операция получения может быть организована асинхронно. Для уменьшения объема коммуникаций устраняются те коммуникации, которые не передают востребованной соседом информации, а вектора и матрицы после сегментации «сжимаются», чтобы в передаваемом векторе присутствовали только востребованные элементы. Чтобы выполнить такие оптимизации необходимо вводить предварительную процедуру, составляющую коммуникационный план. Таким образом, распределённый алгоритм умножения матрицы на вектор разбит на две части: подготовительную, выполняющуюся единожды для данной матрицы, и вычислительную, выполняющую непосредственно вычисление произведения матрицы на вектор для уже подготовленной матрицы. Подготовительная часть не зависит от того, какой вектор будет участвовать в последующей операции и

включает в себя алгоритмы сегментирования матрицы по столбцам и алгоритмы сжатия этих блоков с передачей информации о шаблонах сжатия соседним узлам параллельной системы. Вычислительная часть состоит из операций запуска асинхронных обменов с соседними узлами и асинхронных процедур копирования данных в память графического ускорителя, запуска вычислительной процедуры умножения матрицы на вектор для локального блока матрицы, которые будет выполняться одновременно с этими обменами, и запуска аналогичных процедур для остальных сегментов матрицы по мере получения данных.

Описанная методика построения алгоритма позволяет достичь высокой степени перекрытия различных типов обменов данными с вызовами вычислительных процедур. Как результат, это обеспечило высокий уровень масштабируемости полученного алгоритма.

4. Результаты

4.1 Первичные результаты расчетов в пакете OpenFOAM

Опыт расчетов задач гидродинамики судовых движителей в инженерных CFD-пакетах показывает, что при адекватном выборе расчетных сеток, моделей турбулентности и прочих параметров модели, может быть обеспечена приемлемая при проектировании точность по интегральным характеристикам в пределах 2% от эксперимента [10]. Одной из целей данной работы является оценка возможностей пакета OpenFOAM для решения подобных задач, а также точности получаемых результатов в сравнении с коммерческими инженерными пакетами, в частности Star-CCM. В рамках настоящей работы предполагается проведение методических исследований и расчетов для набора из трех расчетных сеток размером 41, 60 и 99 млн. ячеек, и оценка сеточной сходимости результатов.

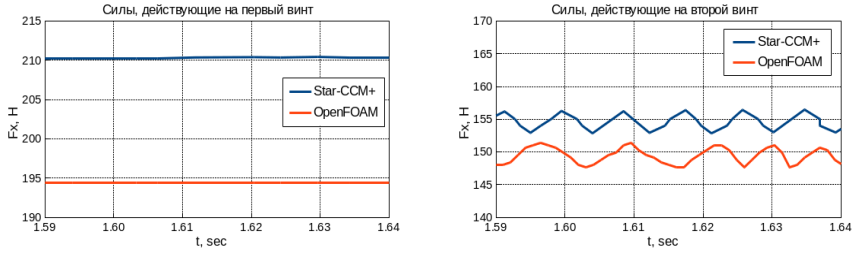


Рис. 3. Графики зависимости осевой силы давления винта, F_x, от времени, t.

На данном этапе работы, однако, пока были получены результаты расчетов только для одной из моделей размером 41 млн. ячеек. Сравнение интегральных характеристик (рис. 3), полученных в расчетах в пакетах OpenFOAM и Star-CCM, демонстрирует расхождение порядка 7%, что

является существенной погрешностью для данного класса задач. Кроме того следует отметить, что период колебаний сил, действующих на второй винт в расчете в пакете OpenFOAM, вообще говоря, оказывается переменным, что вызывает ряд вопросов с точки зрения достоверности полученных результатов. Вместе с тем, к обнадеживающим факторам можно отнести тот факт, что полученные результаты в OpenFOAM, в целом, соотносятся по порядку величины с ожидаемыми значениями, и работа по продолжению методических исследований и оценке сеточной сходимости и влиянию сетки на период колебаний сил имеет смысл. На рис. 4 представлено поле скорости, полученное при расчете в пакете OpenFOAM на сетке 41 млн. ячеек. Картина обтекания иллюстрирует наличие отрывных зон, наиболее обширных в следе за вторым винтом, что качественно соотносится с аналогичными работами [11].

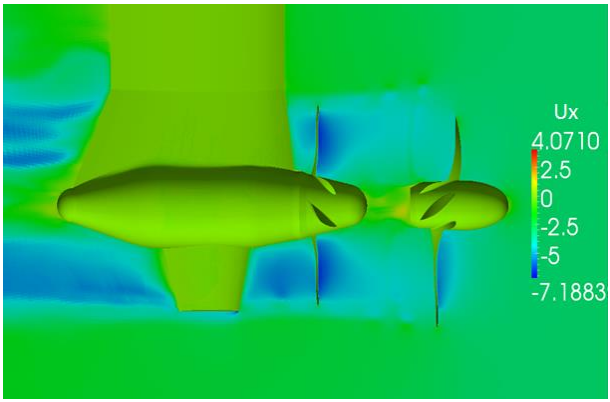


Рис. 4. Поле продольной компоненты скорости.

4.2 Оценка эффективности методов решения СЛАУ в пакете OpenFOAM

Как упоминалось ранее, для аккуратного расчета рассматриваемой задачи необходимо проведение расчета длительностью до сотен тысяч шагов по времени, что может занимать несколько недель. С другой стороны, для оценки эффективности использования различных численных методов и ускорения расчетов нет необходимости проведения полноценного расчета, а достаточно ограничиться небольшим, но репрезентативным интервалом расчета. В данном случае для этих целей был выбран начальный промежуток времени, за который происходит перестроение потока после начала вращения винтов, а именно расчет первых 60 шагов по времени. Как показали расчеты, такие данные являются вполне информативными, поскольку поведение оцениваемых методов решения систем уравнений для давления, возникающих в процессе активного перестроения потока, оказалось существенно

различным. При этом к 60-му шагу основное перестроение потока уже заканчивается, а времена расчета очередного шага по времени перестают сколь-нибудь заметно меняться. Таким образом, эти результаты являются достаточно адекватной оценкой для величины шага интегрирования по времени на протяжении основного расчета.

На вычислительной системе «Ломоносов» была проведена серия расчетов, которая включала сравнительные исследования для конфигураций методов GAMG и PCG+GAMG из пакета OpenFOAM, а также MPI-реализации методов BiCGStab+CAMG из разрабатываемой библиотеки SparseLinSol. В расчетах использовались 128 и 256 ядер, или 16 и 32 узла соответственно. Численные эксперименты показали, что времена расчета одного шага интегрирования по времени для конфигураций методов GAMG и PCG+GAMG в целом демонстрируют сходный характер поведения, но существенным образом зависят от номера шага (рис. 5). На начальных участках графиков, когда происходит интенсивное перестроение потока, наблюдается резкий рост времени расчета с последующим плавным уменьшением по мере стабилизации течения, и этот рост вызван значительным увеличением количества итераций методов при решении СЛАУ для давления. Явно выраженное превышение среднего времени расчета шага наблюдается на протяжении первых 20-30 шагов, однако их расчет занимает дополнительно порядка 10 часов. Рассматриваемый внешний решатель из библиотеки SparseLinSol также на начальном участке затрачивает несколько большее время на решение соответствующих систем уравнений по сравнению с временами в конце тестового диапазона шагов, однако такая разница оказывается много меньшей, в пределах 70-80%.

При выходе на установившийся режим расчета разница во времени решения между двумя конфигурациями методов из пакета OpenFOAM оказывается сравнимой со статистической погрешностью. Шаг по времени на 128 и 256 ядрах рассчитывается за 650 и 470 секунд соответственно, а коэффициент ускорения при удвоении количества расчетных ядер не превышает 1.4. Использованная связка методов BiCGStab+CAMG, несмотря на дополнительные накладные расходы на преобразование данных между форматами представления данных в пакете OpenFOAM и разрабатываемой библиотеке, оказывается более эффективной и с этой точки зрения, обеспечивая времена в 470 и 310 секунд при коэффициенте ускорения чуть более 1.5. Таким образом, использование внешнего решателя позволяет сократить на 10 и более часов время расчета на начальном интервале, и в полтора раза ускорить вычисления на установившемся режиме расчета.

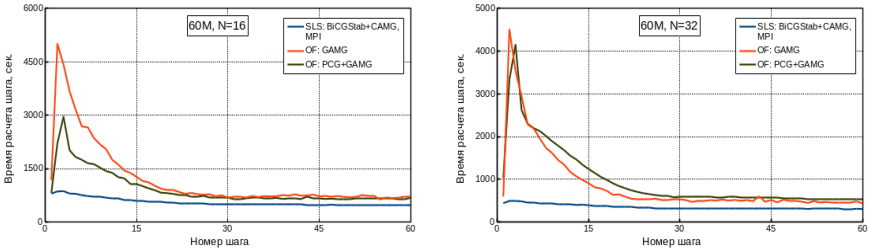


Рис. 5. Графики зависимости времени расчета шага по времени от номера шага. Вычислительная система «Ломоносов»; слева: расчет на 16 узлах, справа: расчет на 32 узлах.

4.3 Применение графических ускорителей для решения СЛАУ

Для оценки производительности и масштабируемости разработанных методов решения СЛАУ в библиотеке *SparseLinSol* проведена серия тестов изолированного этапа решения СЛАУ для матриц, полученных в ходе расчета соответствующих моделей размером 41, 60 и 99 млн. неизвестных в пакете OpenFOAM (этап построения иерархии матриц многосеточного метода в данном случае иключен из измерений). Серия расчетов выполнена на суперкомпьютере “Ломоносов” в трёх конфигурациях библиотеки: 1) с использованием центральных процессоров и MPI-коммуникаций (кривые “CPU: MPI-only” на рис. 6); 2) с использованием центральных процессоров и гибридной схемы коммуникаций MPI+Posix Shared Memory (кривые “CPU: MPI+ShM” на рис. 6); 3) с использованием центральных процессоров и двух графических ускорителей на каждом узле (кривые “GPU” на рис. 6). Варианты использования гибридной схемы демонстрируют значительное превосходство в масштабируемости вплоть до максимального числа узлов, использовавшихся в расчетах. Использование графических ускорителей дает заметный выигрыш на всех тестовых матрицах на диапазоне до 64 узлов, однако масштабируемость алгоритмов, использующих графические ускорители, несколько хуже в сравнении с гибридной схемой MPI+Posix ShM. Для матриц размером 41 и 60 млн. неизвестных преимущества от использования графических ускорителей нивелируются на 128 узлах. Однако, результаты для матрицы большего размера демонстрируют лучший потенциал использования графических ускорителей, и ускорение в диапазоне от 1.4 до 3 раз наблюдается во всем диапазоне использованных узлов.

Для матрицы размером 41 млн. неизвестных выполнено сравнение производительности и масштабируемости с одной из свободно-распространяемых библиотек численных методов *hypre* (кривая “Hypre” на рис. 6). Конфигурация численных методов, которая использовалась в ходе этого сравнения, полностью аналогична: итерационнаый метод BiCGStab и CAMG предобуславливатель с PMIS-алгоритмом агрегирования и

сглаживателем на основе полиномов Чебышёва 2-го порядка. По результатам расчетов получено, что даже MPI-реализация методов из разрабатываемой библиотеки *SparseLinSol* обеспечивает 1.5-2 кратное ускорение в сравнении с MPI-реализацией методов из библиотеки hypre, тогда как использование гибридной модели и графических ускорителей увеличивает коэффициент ускорения до 3-4.

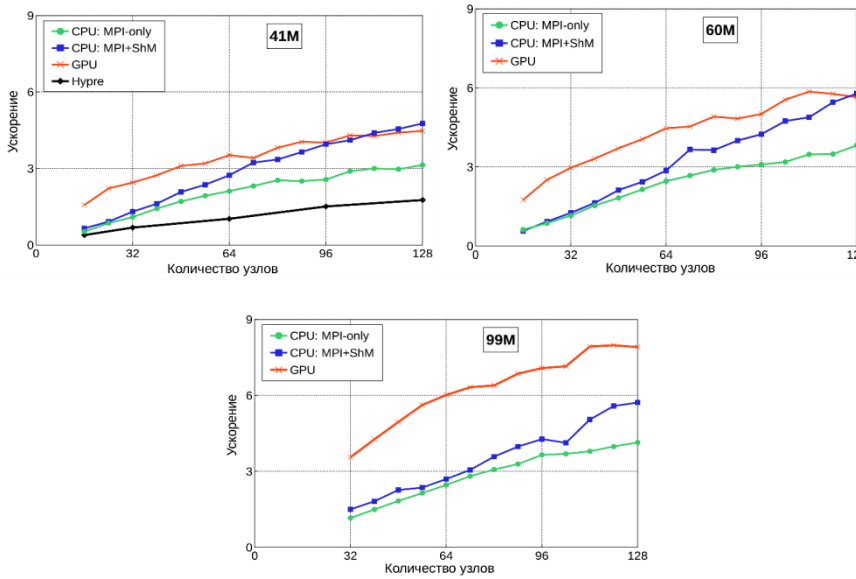


Рис. 6. Графики зависимости ускорения расчетов от количества используемых вычислительных узлов. Вычислительная система «Ломоносов»; результаты нормированы на времена расчета на 32 узлах в режиме «CPU:MPI-only».

Логическим продолжением настоящей работы является проведение серии тестовых расчетов на вычислительных системах, оборудованных графическими ускорителями, для оценки дополнительного эффекта ускорения расчетов, который может быть получен за счет применения гибридных моделей программирования MPI+Posix Shared Memory и графических ускорителей. Отдельный интерес представляет исследование пределов эффективной масштабируемости пакета OpenFOAM в связке с библиотекой *SparseLinSol* для рассматриваемого класса задач, который, как ожидается, должен быть заметно выше, по сравнению со стандартными методами из пакета OpenFOAM.

5. Заключение

В настоящей работе представлены промежуточные результаты проводимой работы по исследованию возможностей пакета OpenFOAM для

моделирования гидродинамических характеристик пары разнонаправленно вращающихся гребных винтов судна. Проведены первичные расчеты задачи на сетке 41 млн. ячеек и сопоставление результатов с аналогичным расчетом в пакете Star-CCM. Хотя интегральные характеристики согласуются в пределах 7%, необходимо проведение дальнейших методических исследований, в том числе изучение вопросов сеточной сходимости результатов.

Рассмотрены вопросы потенциального ускорения расчета рассматриваемого класса задач в пакете OpenFOAM. Показано, что использование разрабатываемой авторами библиотеки SparseLinSol и традиционной MPI-реализации численных методов позволяет добиваться 1.5-кратного ускорения расчета. Приведены результаты масштабируемости этапа solve-части многосеточных методов для гибридной реализации MPI+Posix Shared Memory и реализации, использующей графические ускорители. Продемонстрировано, что применение гибридных моделей и использование графических ускорителей может обеспечивать дополнительное 1.5-3 кратное ускорение по сравнению с MPI-реализацией методов в библиотеке SparseLinSol.

Следующим этапом настоящей работы станет проведение серии тестовых расчетов на вычислительных системах, оборудованных графическими ускорителями, для оценки дополнительного эффекта ускорения расчетов в пакете OpenFOAM, который может быть получен за счет применения гибридных моделей программирования MPI+Posix Shared Memory и графических ускорителей.

Работа выполнена с использованием ресурсов суперкомпьютерного комплекса МГУ имени М.В. Ломоносова и частично поддержана грантом РФФИ 14-01-00295.

Список литературы

[1]. Б. Краснопольский, А. Медведев. О решении систем линейных алгебраических уравнений на многоядерных вычислительных системах с графическими ускорителями. Параллельные вычислительные технологии (ПаВТ'2013): труды международной научной конференции (1–5 апреля 2013 г., г. Челябинск). Челябинск: Издательский центр ЮУрГУ, 2013. 637 с., стр. 409–420.

[2]. Б. Краснопольский, А. Медведев. Алгоритмические особенности создания многосеточного решателя СЛАУ на вычислительных системах с графическими ускорителями. Вестник Нижегородского университета им. Н.И. Лобачевского, серия «Информационные технологии», 2, 2014. стр. 210-217.

[3]. F. Menter. Two-equation Eddy-viscosity Turbulence Models for Engineering Applications. AIAA Journal, 32 (8), 1994. pp. 1598-1605.

[4]. H.A. Van der Vorst. BI-CGSTAB: a Fast and Smoothly Converging Variant of BI-CG for the Solution of Nonsymmetric Linear Systems. SIAM J. Sci. Statist. Comput., 13, 1992. pp. 631–644.

[5]. U. Trottenberg, C.W. Oosterlee, A. Schuller. Multigrid. Academic Press, 2000. 631 p.

[6]. Hypre: a Library of High Performance Preconditioners. https://computation.llnl.gov/casc/linear_solvers/sls_hypre.html. Дата обращения 18.12.2014.

[7]. N. Bell, M. Garland. Efficient Sparse Matrix-vector Multiplication on CUDA. NVIDIA Corporation, Tech. Rep. NVR-2008-004. 2008.
[8]. A. Monakov, A. Lokhmotov, A. Avetisyan. Automatically Tuning Sparse Matrix-vector Multiplication for GPU Architectures. High Performance Embedded Architectures and Compilers, ser. Lecture Notes in Computer Science, 5952, 2010. pp. 111–125.
[9]. J. Kraus, M. Frster, T. Brandes, T. Soddemann, Using LAMA for Efficient AMG on Hybrid Clusters. Computer Science - Research and Development, 28 (2-3), 2013. pp. 211–220.
[10]. М. Лобачев, Н. Овчинников, А. Пустошный. Численное моделирование работы гребного винта в неоднородном потоке. Труды ЦНИИ им. акад. А.Н. Крылова, 333(6), 2009. стр. 5-10.
[11]. M. Peric, V .Bertram. Trends in Industry Applications of CFD for Maritime Flows. 10-th International Conference on Computer and IT Applications in the Maritime Industries. Berlin, Germany, 2011. pp.8-18. http://www.gl-group.com/pdf/Trends_in_Industry_Applications_of_CFD_for_Maritime_Flows.pdf

On application of GPUs for modelling of hydrodynamic characteristics of screw marine propellers in OpenFOAM package

[1,2] *B. Krasnopolsky <krasnopolsky@imec.msu.ru>*
[2] *A. Medvedev <alexey.v.medvedev@gmail.com>*
[1] *A. Chulyunin <chulyu-n@mail.ru>*
[1] *Institute of Mechanics, Lomonosov Moscow State University*
1, Michurinsky ave., Moscow, 119192, Russia.
[2] *JSC T-Services*
113/1, Leninsky ave., Moscow, 117198, Russia.

Abstract. OpenFOAM is a proven engineering tool for applied hydrodynamics numerical modeling which is typically characterized by complex geometries and large grids of 107-108 cells. Since such calculations are often very long and resource-intesive, any way of speeding them up is of high practical interest. Based on one practical problem of a screw propeller characteristics modeling, optimizations to OpenFOAM via the originally developed SLAE solution plugin is proposed. The plugin is based on SparseLinSol (SLS) library, developed by the authors. The library uses Krylov subspace iterative methods with the Classic AMG preconditioner to effectively solve large SLAEs on supercomputers and features original hybrid communications model which implements MPI and Posix Shared Memory combination. The library also is able to utilize NVIDIA GPU accelerators for a significant part of the implemented algorithms. Test results on 128-node computational system equipped with NVIDIA X2070 accelerators show that: (i) OpenFOAM numerical modeling results are close to those achieved with Star-CCM package and experimental results; (ii) developed SLAE solution methods are more robust than those implemented in original OpenFOAM GAMG-based SLAE solver; (iii) hybrid communication model improves solver scalability a lot and the solver scales linearly up to the maximum number of nodes used in current tests; (iv) GPU usage makes calculations 1.4-3 times faster; (v) SLS solver is faster than hypre solver on the same set of implemented methods and test matrices.

Keywords: multigrid methods, GPU, marine screw propellers, scalability, OpenFOAM package.

References

[1]. B. Krasnopol'skij, A. Medvedev. O reshenii sistem linejnykh algebraicheskikh uravnenij na mnogojadernykh vychislitel'nykh sistemakh s graficheskimi uskoriteljami [On solution of systems of linear algebraic eqautions on multicore systems with GPU accelerators]. Parallel'nye vychislitel'nye tekhnologii (PaVT'2013): trudy mezhdunarodnoj nauchnoj konferencii (1–5 aprelja 2013, Cheljabinsk) [Parallel Computing technologies (PaCT'2013): proceedings of international scientific conference (April 1–5, 2013, Chelyabinsk)]. Cheljabinsk: Izdatel'skij centr ÜUrGU, 2013. 637 p., pp 409–420 (in Russian).

[2]. B. Krasnopol'skij, A. Medvedev. Algoritmicheskie osobennosti sozdanija mnogosetochnogo reshatelja SLAU na vychislitel'nykh sistemakh s graficheskimi uskoriteljami [Algorithmic aspects of development the multigrid slae solver for computer systems with GPU accelerators]. Vestnik Nizhegorodskogo universiteta im. N.I. Lobachevskogo, serija «Informacionnye tekhnologii» [Vestnik of Lobachevsky State University of Nizhni Novgorod, "Information Technology" series], 2, 2014. pp. 210-217 (in Russian).

[3]. F. Menter. Two-equation Eddy-viscosity Turbulence Models for Engineering Applications. AIAA Journal, 32 (8), 1994. pp. 1598-1605.

[4]. H.A. Van der Vorst. BI-CGSTAB: a Fast and Smoothly Converging Variant of BI-CG for the Solution of Nonsymmetric Linear Systems. SIAM J. Sci. Statist. Comput., 13, 1992. pp. 631–644.

[5]. U. Trottenberg, C.W. Oosterlee, A. Schuller. Multigrid. Academic Press, 2000. 631 p.

[6]. Hypre: a Library of High Performance Preconditioners. https://computation.llnl.gov/casc/linear_solvers/sls_hypre.html. Дата обращения 18.12.2014.

[7]. N. Bell, M. Garland. Efficient Sparse Matrix-vector Multiplication on CUDA. NVIDIA Corporation, Tech. Rep. NVR-2008-004. 2008.

[8]. A. Monakov, A. Lokhmotov, A. Avetisyan. Automatically Tuning Sparse Matrix-vector Multiplication for GPU Architectures. High Performance Embedded Architectures and Compilers, ser. Lecture Notes in Computer Science, 5952, 2010. pp. 111–125.

[9]. J. Kraus, M. Frster, T. Brandes, T. Soddemann, Using LAMA for Efficient AMG on Hybrid Clusters. Computer Science - Research and Development, 28 (2-3), 2013. pp. 211–220.

[10]. M. Lobachev, N. Ovchinnikov, A. Pustoshnyj. Chislennoe modelirovanie raboty grebnogo vinta v neodnorodnom potoke [Numerical modeling of marine propeller in a nonuniform flow]. Trudy CNII im. akad. A.N. Krylova [Proceedings of Krylov Shipbuilding Research Institute Acad. A.N. Krylov], 333(6), 2009. pp. 5-10 (in Russian).

[11]. M. Peric, V .Bertram. Trends in Industry Applications of CFD for Maritime Flows. 10-th International Conference on Computer and IT Applications in the Maritime Industries. Berlin, Germany, 2011. pp. 8-18. http://www.gl-group.com/pdf/Trends_in_Industry_Applications_of_CFD_for_Maritime_Flows.pdf

Расчет распада произвольного разрыва в двухскоростном потоке с несжимаемыми компонентами

Б.Л. Канцырев <Boris.Kantsyrev@mail.ru>
Институт Океанологии им. П.П.Ширшова,
Россия, Москва, Нахимовский просп.36

Аннотация. Представлен анализ влияния вида пульсационных слагаемых в системе осредненных уравнений гидродинамики пузырькового потока с несжимаемыми фазами на волновые свойства модели. Получены соотношения, определяющие решение задачи о распаде произвольного разрыва в газожидкостном пузырьковом потоке с несжимаемыми компонентами.

Ключевые слова: пульсационные слагаемые; характеристика; автомодельное решение; решение Римана; распад произвольного разрыва**.**

1. Введение

В настоящее время в литературе уже устоялись представления о виде системы уравнений двухскоростного движения гетерогенных сред, о структуре слагаемых, входящих в неё. Однако установление соотношений, которые определяют вид коэффициентов при дифференциальных слагаемых, до сих пор представляет собой еще не решенную и актуальную задачу. Действительно, указанными коэффициентами определяется тип и волновые свойства указанной системы. Недостаток информации о коэффициентах часто приводит к ухудшению расчетной модели, несмотря на более детальный учёт межфазных сил и совершаемой ими работы. Поэтому в данной работе рассмотрено получение соотношений, уточняющих коэффициенты при дифференциальных слагаемых в уравнениях импульса и энергии гетерогенного пузырькового потока. Это дает возможность конкретизировать волновые свойства системы осредненных уравнений, т.е. характеристические скорости и автомодельные решения типа центрированной волны и получить решение задачи Римана, необходимое для построения численной схемы, аналогичной схеме С.Годунова [1] . Рассмотрим режим движения с достаточно большими характерными временами, для которых можно пренебречь сжимаемостью фаз, а скорость изменения размера пузырька по порядку величины не превышает

относительной скорости фаз [2, гл.6, §6]. Прежде, чем записать систему уравнений двухскоростного движения газожидкостного потока учтём, что при выводе осреднённых уравнений многофазной гидродинамики получаются дополнительные слагаемые, соответствующие вкладу пульсационных напряжений, аналогичных рейнольдсовым напряжениям в турбулентных потоках. Действительно, в соответствии с [3], (гл 1, §1), систему уравнений гидродинамики для фазы сплошной среды , обозначенной индексом «i» можно представить в виде :

$$\rho_i^{0'}\frac{\mathrm{d}_i E_i^{'}}{\mathrm{d}t}=\frac{\partial}{\partial t}\rho_i^{0'}E_i^{'}+\nabla^{'k}\rho_i^{0'}E_i^{'}v_i^{'k}=\nabla^{'k}\vartheta_i^{'k}+\rho_i^{\circ'}f^{'} \qquad (1)$$

где

$$E_i^{'}=1;\quad v_i^{'};\quad e_i^{'}+\frac{1}{2}\left(v_i^{'}\right)^2$$

$$\vartheta_i^{'k}=0;\quad \sigma_i^{'k};\quad \sigma_i^{'k}v_i^{'}-q_i^{'k}$$

$$f^{'}=0\ ;\quad g_i\ ;\qquad g_i\ v_i^{'}$$

При осреднении каждое из слагаемых в уравнениях заменяется осреднённым. Вводя обозначение для отклонения от среднего значения :

$$\Delta E_i^{'}=E_i^{'}-E_i\,, \qquad (2)$$

можно показать, как это было сделано в ([1],гл 1, §2), что среднее значение

$$\left\langle\rho_i^{\circ'}E_i^{'}v_i^{'k}\right\rangle_i=\left\langle\rho_i^{\circ'}\right\rangle\left\langle E_i^{'}\right\rangle\left\langle v_i^{k'}\right\rangle+\left\langle\rho_i^{\circ'}\Delta E_i^{'}\Delta v_i^{'k}\right\rangle_i \qquad (3)$$

Второе слагаемое в правой части (3) соответствует пульсационным добавкам.

Таким образом, в осреднённой системе уравнений под знаком дифференцирования оказываются не только осреднённые значения переменных, но и пульсационные. Их расчёт в общем случае представляет собой весьма сложную задачу, поэтому в данной работе рассмотрен специальный «ламинарный режим» движения дисперсной смеси. В таком режиме отсутствует хаотичное движение дисперсной фазы, а пульсации скоростей несущей (жидкой) фазы определяются разностью осреднённых скоростей фаз. Как было показано в [4, гл 16], этот режим действительно реализуется при не слишком больших скоростях потоков. В данной работе рассмотрено влияние указанных пульсационных слагаемых кинетической энергии и тензора напряжений несущей фазы на волновые свойства системы уравнений пузырькового потока. Кинетическая энергия мелкомасштабных движений и пульсационная составляющая тензора поверхностных напряжений, соответствующая пространственно одномерному осредненному

уравнению движения жидкой фазы в указанном ламинарном режиме могут быть представлены [3] соответственно в виде;

$$k_1 = 0.5\alpha_2\chi(\alpha_2)\upsilon^2$$
$$\Pi_1 = -0.5\alpha_2\psi(\alpha_2)\ \upsilon^2 \qquad (4)$$

где α_2 – объемное газосодержание, u - относительная скорость фаз $u = V_2 - V_1$, V_2 , V_1- соответственно макроскопические (осредненные) скорости дисперсной и несущей фазы, χ и ψ -т.н. пульсационные коэффициенты, которые являются функциями объемного газосодержания α_2. При $\alpha_2 > 0$ они рассматриваются, как искомые величины.

2. Система уравнений

Запишем уравнения баланса импульса и энергии для двухфазного потока в целом с учетом сделанных выше предположений.

$$\frac{\partial\left(\rho_1^\circ\alpha_1 V_1 + \rho_2^\circ\alpha_2 V_2\right)}{\partial t} + \frac{\partial\left(p_1\alpha_1 + p_2\alpha_2 + \rho_1^\circ\alpha_1 V_1^2 + \rho_2^\circ\alpha_2 V_2^2 - \rho_1^\circ\Pi_1\right)}{\partial z} = F_{ext} \qquad (5)$$

$$\frac{\partial\left(\rho_1^\circ\alpha_1\left(e_1 + \frac{1}{2}V_1^2 + k_1\right) + \rho_2^\circ\alpha_2\left(e_2 + \frac{1}{2}V_2^2\right)\right)}{\partial t} +$$
$$+ \frac{\partial\left(\rho_1^\circ\alpha_1 V_1\left(e_1 + \frac{1}{2}V_1^2 + k_1\right) + \rho_2^\circ\alpha_2 V_2\left(e_2 + \frac{1}{2}V_2^2\right) - C_1\right)}{\partial z} = F_{ext}V \qquad (6)$$

Индекс 1 относится к жидкой (несущей) фазе, а 2- к дисперсной фазе.

В дальнейшем приняты обозначения: $d_к/dt = \partial/\partial t + V_к\,\partial/\partial z$, к=1,2. $\alpha_1 + \alpha_2 = 1$,

$p = p_1\alpha_1 + p_2\alpha_2$ – давление в потоке , $p_1 - p_2 = \Lambda\rho_1^\circ u^2$ где $\Lambda = \frac{1}{4}\left(1 - \frac{\alpha_2}{\alpha_1}\right)$

– соответственно, разность между осредненными величинами давления в первой фазе и на поверхности пузырька постоянного радиуса [1, гл.3, стр 130], $W = \alpha_1 V_1 + \alpha_2 V_2$,

$\sigma^1 = -p + \rho_1^\circ\Pi_1$ $\qquad$ $C_1 = -p_1\alpha_1 V_1 - p_2\alpha_2 V_2 + V_2\rho_1^\circ\Pi_1$

-напряжения поверхностных сил и работа поверхностных сил с учетом пульсационных составляющих. В дальнейшем предположим, что фазы

несжимаемы, фазовый переход не учитывается (газо-жидкостная среда). Уравнения баланса тепловой энергии имеют следующий вид:

$$\begin{aligned}\rho_1^{\circ}\alpha_1\frac{\mathrm{d}_1 e_1}{\mathrm{d}t}&=\chi_1 K_{\mu}\alpha_1\alpha_2\left(V_2-V_1\right)^2+Q_{21}\\ \rho_2^{\circ}\alpha_2\frac{\mathrm{d}_2 e_2}{\mathrm{d}t}&=\chi_2 K_{\mu}\alpha_1\alpha_2\left(V_2-V_1\right)^2+Q_{12}\end{aligned}\qquad(7)$$

Где χ_1 , χ_2 - коэффициенты, учитывающие доли работы вязких сил, сосредоточенных на межфазной границе, приходящиеся на жидкую и газовую фазы. Уравнения неразрывности для фаз имеют вид:

$$\begin{aligned}\frac{\partial\alpha_1}{\partial t}+\frac{\partial\alpha_1 V_1}{\partial z}&=0,\\ \frac{\partial\alpha_2}{\partial t}+\frac{\partial\alpha_2 V_2}{\partial z}&=0\end{aligned}\qquad(8)$$

В соответствии с рассуждениями, изложенными в [4,§ 139], уравнения двухскоростного движения не должны противоречить законам сохранения импульса и полной энергии, а условие непротиворечивости позволяет конкретизировать вид уравнений движения.. Условием непротиворечивости уравнений движения и законов сохранения в данном случае оказывается соотношение, отражающее тот факт, что притоки тепла от фаз к межфазной границе равны по величине и противоположны:

$$Q_{12}+Q_{21}=0$$

При этом (5-6) с учётом (7,8) можно преобразовать к следующему виду [6.7];

$$\begin{aligned}&\frac{\partial\left(\rho_1^0\alpha_1 v_1\right)}{\partial t}+\frac{\partial\left(\rho_1^0\alpha_1 v_1^2-\alpha_1\sigma^1\right)}{\partial z}=-p\frac{\partial\alpha_2}{\partial z}-\chi\rho_1^0\alpha_1\alpha_2\left(\frac{\mathrm{d}_1 V_1}{\mathrm{d}t}-\frac{\mathrm{d}_2 V_2}{\mathrm{d}t}\right)+\\ &+\rho_1^0\Phi_{\alpha}u^2\frac{\partial\alpha_2}{\partial z}+\rho_1^0\Phi_{\upsilon}u\frac{\partial u}{\partial z}+F_1\end{aligned}\qquad(9)$$

$$\begin{aligned}&\frac{\partial\left(\rho_2^0\alpha_2 v_2\right)}{\partial t}+\frac{\partial\left(\rho_2^0\alpha_2 v_2^2-\alpha_2\sigma^1\right)}{\partial z}=p\frac{\partial\alpha_2}{\partial z}+\chi\rho_1^0\alpha_1\alpha_2\left(\frac{\mathrm{d}_1 V_1}{\mathrm{d}t}-\frac{\mathrm{d}_2 V_2}{\mathrm{d}t}\right)-\\ &-\rho_1^0\Phi_{\alpha}u^2\frac{\partial\alpha_2}{\partial z}-\rho_1^0\Phi_{\upsilon}u\frac{\partial u}{\partial z}+F_2\end{aligned}\qquad(10)$$

где $\phi_{\alpha} = \frac{\mathrm{d}\left(\alpha_1\alpha_2\left(\psi-\Lambda\right)\right)}{\mathrm{d}\alpha_2} - \alpha_2\psi - \frac{\alpha_1}{2}\left[\left(\chi - 3\alpha_2\right) + \alpha_1\alpha_2\frac{\mathrm{d}\chi}{\mathrm{d}\alpha_2}\right]$

$\phi_u = \alpha_1\alpha_2\left[3\left(\psi - \Lambda\right) - \frac{\alpha_1}{2}\left(3\chi + \alpha_2\frac{\mathrm{d}\chi}{\mathrm{d}\alpha_2}\right)\right]$

Первое слагаемое в правых частях (3.2.5), (3.2.6) представляет межфазную силу Рахматулина. Учет кинетической энергии мелкомасштабных движений в уравнениях (5,6) обеспечивает в (9,10) учет силы присоединенных масс;

$$\chi\left(\alpha_2\right)\rho_1^{\circ}\alpha_1\alpha_2\left(\frac{\mathrm{d}_1 V_1}{\mathrm{d}t} - \frac{\mathrm{d}_2 V_2}{\mathrm{d}t}\right)$$

причем полученное в [1] значение χ =0.5 для α_2=0 соответствует известному из литературы предельному значению коэффициента присоединенных масс для уединённой сферы. Слагаемые в правых частях (9) и (10), пропорциональные Φ_α и Φ_u соответствуют межфазным силам, обусловленным мелкомасштабными движениями и коллективными взаимодействиями дисперсных частиц с несущим потоком]. F_1, F_2 в правой части (9),(10) представляют внешние объемные силы и межфазное взаимодействие, обусловленное вязкостью:

$$F_1 = \rho_1^{\circ}\alpha_1 g_z - F_{12}, \quad F_2 = \rho_2^{\circ}\alpha_2 g_z + F_{12}$$
$$F_{12} = -\alpha_1\alpha_2 K_{\mu} U \qquad (11)$$

.Представляя скорость каждой из компонент в зависимости от полного объёмного расхода W и проскальзывания u после исключения из (9), (10) слагаемых, содержащих $\partial P/\partial z$, получим уравнение для проскальзывания фаз:

$$\frac{\partial u}{\partial t} + \left(W + uK_{\mu}\right)\frac{\partial u}{\partial z} + u^2 K_{\alpha}\frac{\partial \alpha_2}{\partial z} = F \qquad (12)$$

где

$$K_u = \frac{\rho_*}{\rho_1\left(1+\chi\frac{\rho_*}{\rho_2}\right)}\left\{\alpha_1 - \alpha_2\frac{\rho_1}{\rho_2} + \chi\left(\alpha_1 - \alpha_2\right)\frac{\rho_1}{\rho_2} + \frac{\Phi_u}{\alpha_1\alpha_2}\frac{\rho_1}{\rho_2}\right\}$$

$$K_\alpha = \frac{\rho_*}{\rho_2\left(1+\chi\frac{\rho_*}{\rho_2}\right)}\left\{\frac{\left(\Phi_\alpha+\alpha_2\psi\right)}{\alpha_1\alpha_2}-\chi-\frac{\rho}{\rho_1}\right\}$$

$$\frac{1}{\rho_*}=\left(\frac{\alpha_1}{\rho_1}+\frac{\alpha_2}{\rho_2}\right),$$

$$F=-\left[K_\mu U+g^*\left(\rho_1-\rho_2\right)\right]\frac{\rho_*}{\left(\rho_1\rho_2\left(1+\chi\frac{\rho_*}{\rho_2}\right)\right)},\quad g^*=g_z-\frac{\mathrm{d}W}{\mathrm{d}t}$$

Рассмотрим уравнение (12) совместно с уравнениями неразрывности для жидкой и газовой фазы.

$$\frac{\partial\alpha_1}{\partial t}+\frac{\partial\alpha_1 V_1}{\partial z}=0, \tag{13}$$

$$\frac{\partial\alpha_2}{\partial t}+\frac{\partial\alpha_2 V_2}{\partial z}=0 \tag{14}$$

При отсутствии сжимаемости фаз из (13), (14) следует условие независимости полного объемного расхода W от координаты (что позволяет рассматривать W как граничное условие – заданную функцию времени) и уравнение для расчёта объемного газосодержания:

$$\partial\alpha_2/\partial t + \partial(\alpha_2 V_2)/\partial z = 0. \tag{15}$$

Учитывая, что $V_2 = W + \alpha_1 U$, получим из (14)

$$\partial\alpha_2/\partial t + \alpha_1\alpha_2\,\partial U/\partial z + [W+(\alpha_1-\alpha_2)U]\partial\alpha_2/\partial z = 0. \tag{16}$$

Система уравнений (12) и (16) замкнута. В дальнейшем для определённости будем считать, что W=const. Поскольку конкретизация любых моделей тесно связана с обобщением экспериментальных данных, в настоящей работе был рассмотрен один из важных режимов течения двухскоростного потока с несжимаемыми фазами- режим непрерывных волн, соответствующий известной модели «потока-дрейфа» [5]. В основу этой модели положены эмпирические соотношения, связывающие относительную скорость фаз и газосодержание, вида $u=u_0(\alpha_2)$. Подставляя эмпирическое соотношение в уравнение неразрывности (14), можно привести его к виду уравнения переноса

$$\frac{\partial \alpha_2}{\partial t} + u_{\mathrm{DF}} \frac{\partial \alpha_2}{\partial z} = 0 \qquad (17)$$

где скорость распространения волны

$$u_{\mathrm{df}} = W + u(\alpha_1 - \alpha_2) + \alpha_1 \alpha_2 \left(\frac{\mathrm{d}u}{\mathrm{d}\alpha} \right)$$

Как известно, в [3,5] на основании ряда сопоставлений результатов расчётов и экспериментальных данных было показано, что указанная модель правильно описывает квазистационарные волновые процессы в двухфазных потоках. Существенно, что эмпирические соотношения для проскальзывания позволяют моделировать не только движение гладких волн, но и скачков газо- и паросодержания, т.н. «непрерывные ударные волны» (Г. Уоллис). Более того, в [3, Т2] с позиций модели дрейфа был успешно проведён анализ такого существенно нестационарного процесса, как распад произвольного разрыва в потоке с несжимаемыми фазами. Таким образом, в рамках указанной модели правильный результат расчёта может быть получен сравнительно просто. С другой стороны, более детальное двухскоростное моделирование, когда каждой фазе соответствует отдельное дифференциальное уравнение движения, не всегда приводит к правильному результату даже в том случае, когда система уравнений газожидкостного потока гиперболична. Например, первоначально разрывное решение может «размываться» со временем и модель не описывает движение скачка. С учётом этих обстоятельств, представляется целесообразным определить параметры системы уравнений двухскоростного движения исходя из соответствия с волновыми свойствами модели дрейфа в таких режимах течений, когда указанные эмпирические соотношения выполняются. Оказывается, что зависимости $\chi(\alpha_2)$ и $\psi(\alpha_2)$ можно единственным образом определить так, чтобы эмпирические соотношения для проскальзывания были частными решениями системы уравнений двухскоростного движения.

3. Волновые свойства системы уравнений гидродинамики пузырькового потока с несжимаемыми фазами

Существенно, что при $\chi(\alpha_2)$ и $\psi(\alpha_2)$, определённых таким образом уравнения (12), (16) можно привести к дивергентному виду:

$$\frac{\partial T}{\partial t} + \frac{\partial T V_2}{\partial z} = \mathrm{const} \frac{\alpha_2}{u_{\mathrm{w}}(\alpha_2)} \left[g_z (\rho_1^{\circ} - \rho_2^{\circ}) + K_{\mu} U \right] \qquad (18)$$

$$\frac{\partial \alpha_2}{\partial t}+\frac{\partial \alpha_2 V_2}{\partial z}=0 \qquad T=\text{const}\,\alpha_2\left(\frac{U}{U_{\text{w}}(\alpha_2)}\right)$$

Данная форма записи компактна и удобна для реализации расчетного метода С.К. Годунова. Следует отметить, что проведённый в **[21]** анализ также привёл авторов к выводу о совпадении одной из конвективных характеристик системы уравнений многоскоростной гидродинамики со скоростью $\boldsymbol{V_{df}}$, и к целесообразности такой конкретизации коэффициентов при дифференциальных слагаемых, моделирующих межфазные силы, которые обеспечивали бы дивергентный вид уравнений импульса.

4. Расчет распада произвольного разрыва в двухскоростном потоке с несжимаемыми компонентами

Решение задачи о распаде произвольного разрыва (задача Римана) в среде с двумя скоростями создает возможность построения точных численных схем типа схемы Годунова [2] для интегрирования системы уравнений гидродинамики газо- и парожидкостного потока. На практике точное решение нелинейной задачи Римана актуально при моделировании течений с большими градиентами параметров потока, но с другой стороны –весьма сложно. Поэтому в настоящее время разработаны численные методы с линейной аппроксимацией этого решения. Например, в [23-25] представлен основанный на линейной аппроксимации. численный метод P. Roe для двухфазных потоков. Другими словами, решение задачи Римана, используемое на промежуточном этапе вычислений, получается в линеаризованной форме. При этом не всегда гарантируется успешное решение расчетных проблем, обусловленных наличием больших градиентов газосодержания , скоростей, и других параметров течения. Один из перспективных путей решения этих проблем - построение решения задачи Римана, учитывающего движение волн Римана, и разрывов. Например, в работе [22] рассмотрена не гиперболическая система двух уравнений газо-жидкостного потока с несжимаемыми компонентами и получено решение нелинейной задачи Римана с учетом движения контактных разрывов и волн разрежения - сжатия. Основные трудности на этом пути, по-видимому, заключаются в в том, чтобы правильно учесть влияние скоростной неравновесности на решение задачи о распаде произвольного разрыва гидродинамических параметров (давления, скорости, энергии , объемной доли) для каждой из компонент газожидкостного потока. Конкретизация соотношений, определяющих волновые свойства системы уравнений (12),(16), позволяет перейти к построению решения задачи о распаде произвольного разрыва. Поскольку в отсутствие сжимаемости компонент можно отделить задачу нахождения поля скоростей от задачи расчета распределения давления, последнее определяется из решения для

скоростей. Полный объемный расход предполагается известной постоянной величиной. Таким образом, также как и в **[16]** , исходными параметрами считаются проскальзывание u и газосодержание α_2, заданные по обе стороны от разрыва. В дальнейшем терминами ‘волна сжатия’ и ‘волна разрежения’ обозначаются центрированные волны, в которых изменение плотности среды происходит за счет изменения газосодержания. Кроме того:

1.-в соответствии с классической постановкой рассматривается движение среды в отсутствие объемных и вязких сил. Поэтому система уравнений примет вид:

$$\frac{\partial\left(\rho_1^{\circ}\alpha_1 V_1+\rho_2^{\circ}\alpha_2 V_2\right)}{\partial t}+\frac{\partial\left(p+\rho_1^{\circ}\alpha_1 V_1^2+\rho_2^{\circ}\alpha_2 V_2^2-\rho_1^{\circ}\Pi_1\right)}{\partial z}=0 \tag{19}$$

$$\frac{\partial T}{\partial t}+\frac{\partial T V_2}{\partial z}=0 \tag{20}$$

$$\frac{\partial\left(\rho_2^{\circ}\alpha_2\right)}{\partial t}+\frac{\partial\left(\rho_2^{\circ}\alpha_2 V_2\right)}{\partial z}=0 \tag{21}$$

2.- в соответствии с данным в **[1]** обоснованием структуры решения предполагается, что первоначальный разрыв распадается на комбинацию вторичных разрывов и центрированных волн, для которых при $\lambda=z/t$ все зависимые переменные считаются функциями аргумента λ. Рассмотрим автомодельные решения системы (19)-(21) типа центрированной волны. Характеристикам $\boldsymbol{\lambda_1}$ и $\boldsymbol{\lambda_2}$ соответствует решение:

$$\mathrm{d}U/\mathrm{d}\alpha_2 = U\,\Phi_{\mathrm{ui}}(\alpha_2), \tag{22}$$

которое получается из (20)-(21) , а также соотношение, определяющее давление, следующее из (19):

$$\mathrm{d}P/\mathrm{d}\alpha_2 = \rho_1\,U^2\,\Phi_{\mathrm{pi}}(\alpha_2), \tag{23}$$

Φ_{pi} **(i=1,2 –** соответствует номеру характеристики**)** -функция газосодержания, определяемые из (19)- (21). Соотношение, определяющее $\Phi_{\mathrm{ui}}(\alpha_2)$, для $\boldsymbol{\lambda=\lambda_1=\lambda_{df}}$ имеет вид:

$$\Phi_{\mathrm{u1}}(\alpha_2) = (1/U_{\mathrm{w}})\,\mathrm{d}U_{\mathrm{w}}/\mathrm{d}\alpha_2 \tag{24}$$

а для $\boldsymbol{\lambda=\lambda_2}=V_2$

$$\Phi_{u2}(\alpha_2) = 1/\alpha_1 \qquad (25)$$

По терминологии, принятой в **[5]**, данные волновые решения соответствуют динамическим волнам, поскольку они обусловлены силами, пропорциональными градиентам параметров потока (u и α_2).

Зная зависимости (22, 23), можно определить также изменение давления в гладких волнах, соответствующих характеристикам. На рис 1 они представлены в виде зависимостей давления от плотности среды. Плотность первой фазы 500 кг /м^3, второй -10 кг /м^3 . По своему смыслу эти зависимости аналогичны адиабатам Пуассона. Для расчета соотношений на скачке, исходя (19)- (21) можно получить также соотношения на разрыве в газожидкостном пузырьковом потоке с несжимаемыми компонентами. В теории одномерных двухфазных течений [5] подобные разрывы называются «непрерывными ударными волнами». Рассматривая всевозможные конфигурации вторичных волн и разрывов, можно показать [7] ,что задача о распаде произвольного разрыва имеет единственное решение.

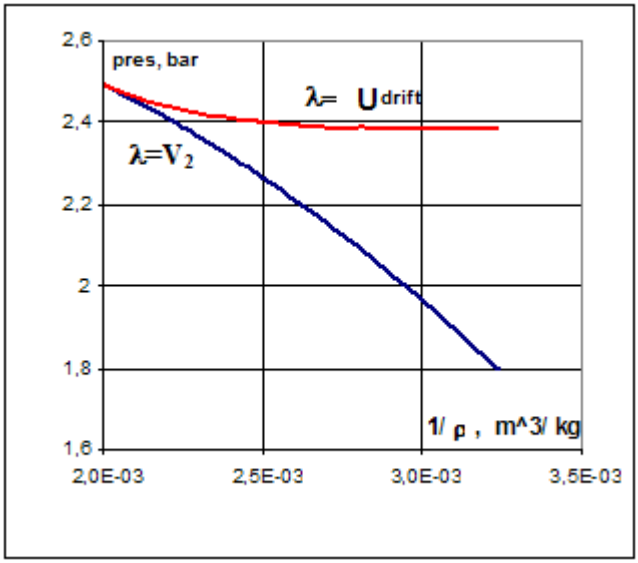


Рис 1.

На рис 2 представлено распределение газосодержания по координате для различных моментов времени, полученное в соответствии с представленным алгоритмом расчета. Кривая **‘0’** представляет собой распределение газосодержания в начальный момент времени. Слева от разрыва газосодержание α_2 равно 0.05 , справа – 0.15. Кривые, соответствующие расчетным вариантам a и b соответствуют распаду разрыва с образованием двух гладких волн, причем на кривой a имеется область с

нулевым газосодержанием между фронтами волн . Кривой с соответствует движение волны и разрыва. Кривая d соответствует решению, построенному в виде комбинации двух разрывов. Следует отметить, что логика представленной схемы распада разрыва соответствует задаче Римана для совершенного газа. Действительно, в классическом решении задачи Римана, как известно, возможно образование двух волн разрежения, волны разрежения и разрыва, а также образование двух разрывов. На основе полученного алгоритма расчета распада произвольного разрыва в потоке с несжимаемыми фазами был создан расчетный код, использующий решение задачи Римана, полученное в [4].

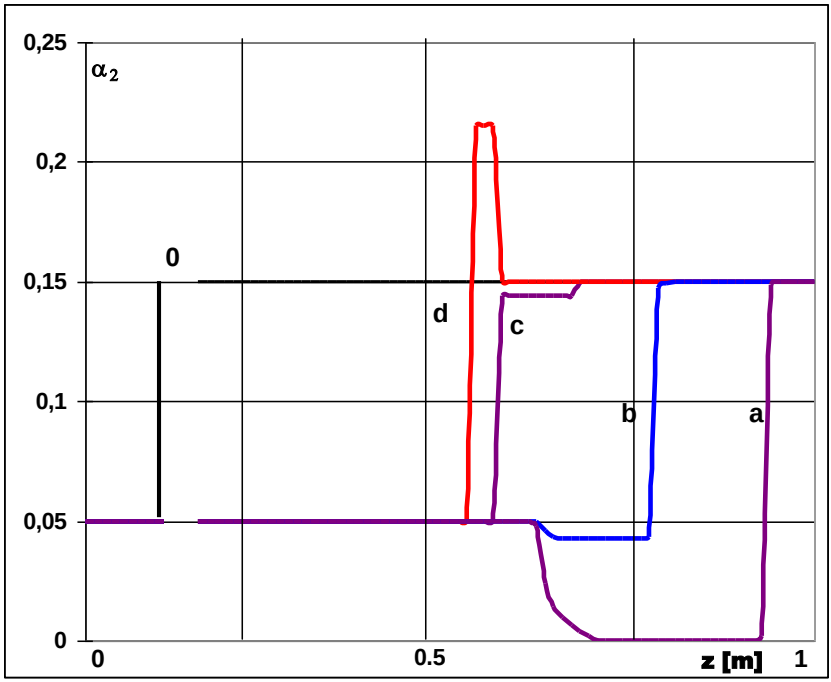


Рис 2.

Распределения газосодержания по координате для двух фиксированных моментов времени. В момент времени **t=0** (распределение **‘0’**) слева от разрыва α_2**=0.05** и **U=1 м/с,** -справа α_2**=0.15.** Кривые **a-d** соответствуют моменту времени **t=1** с. и различным начальным значениям проскальзывания *U* справа от разрыва. Соответственно для кривых **a**- 1.5 м/с , **b**- 1.2 м/с, **c**- 0.8 м/с, **d** – 0.6 м/с.

Результат следующего тестирования показан на рис 3,. где представлены распределения газосодержания для нестационарного противоточного

течения в вертикальном канале с закрытым дном. Верхнее поперечное сечение канала моделируется, как граница с ёмкостью бесконечно большого объема, заполненной жидкостью Продольная координата увеличивается снизу вверх Видно, что граница растущего уровня жидкости не размывается в течение всего процесса. Кроме того, видно, что стекающий вниз поток постепенно сжимается вследствие ускорения на участке соответствующем продольной координате, большей 0.7 м

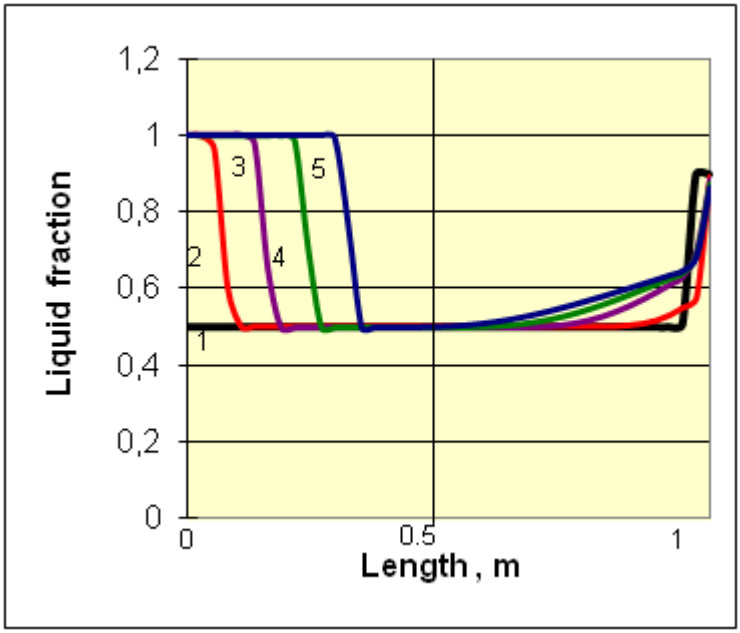


Рис 3. t (сек)= 1.- 0; 2.- 2; 3.- 4; 4.- 6; 5.- 8;

5. Заключение

Получена система уравнений гидродинамики для газожидкостного пузырькового потока, с волновыми свойствами, определяемыми вкладом пульсационных слагаемых. Представлена гидродинамическая модель пузырькового потока, в рамках которой одной из характеристических скоростей газожидкостной дисперсной смеси следует считать скорость переноса возмущений газосодержания, соответствующую модели потока-дрейфа, а другой- скорость пузырьков в потоке.. Осредненная система уравнений имеет правильные волновые свойства только при учете различия давлений компонент. Получены автомодельные решения системы уравнений, соответствующие ее характеристикам. Как следствие данной системы уравнений, записанных в форме законов сохранения, получены соотношения, определяющие ее разрывные решения. Созданная расчётная

программа позволяет решать представленную систему уравнений гидродинамики двухскоростного пузырькового потока с помощью метода С.К. Годунова, адаптированного для численного моделирования двухфазной среды.

Список литературы

[1]. Годунов С.К, Численное решение многомерных задач газовой динамики, М.Наука, 1976, 400 стр
[2]. Нигматулин Р.И. Механика сплошной среды, М. Геотар, 2014, 639 стр
[3]. Р.И. Нигматулин. Основы механики гетерогенных сред. М. Наука, 1978, 336 стр
[4]. Л.Д. Ландау, Е.М. Лифшиц, Гидродинамика Т6, М. Наука, 711стр
[5]. Г. Уоллис, Одномерные двухфазные течения, М. Мир, 1972, 440 стр
[6]. B.L. Kantsyrev. Riemann Problem When Modeling Dual-Speed Bubble Flow.//Proceedings of ICAPP'06, Reno, NV USA, June 4-8, 2006, Paper 6016, pp 1526-1529
[7]. B.L. Kantsyrev. Modification of Godunov Computation Method for Modeling Non-Stationary Gas-Liquid Flow.// Proceedings of ICAPP'07.Paper 7061, Nice Acropolis, France,May 13-18, 2007

Calculation of the disintegration of any break in the flow of a two-speed two-phase incompressible

B.L. Kantsyrev <Boris.kantsyrev@mail.ru>
Institute of Oceanology
Russia,Moscow, Nahimovsky broad street 36

Abstract. It is proposed to modify the Godunov computation method for modeling the gas-liquid bubble flow. To have a numerical scheme, effect of the coefficients – at the differential constituents in the hydrodynamic averaged equations set – on the hydrodynamic wave properties is discussed. Specific solution of Riemann problem has been obtined.

Keywords: pulsation terms; characteristic; self-determination; Riemann solution; decay of an arbitrary discontinuity.

References

[1]. Godunov, S. K., A Difference Scheme for Numerical Solution of Discontinuous Solution of Hydrodynamic Equations, *Math. Sbornik,* **47**, 271-306, translated US Joint Publ. Res. Service, JPRS 7225 Nov. 29, 1960
[2]. R.I. Nigmatulin Continuum Mechanics, M. Geotar, 2014, 639 pages. (in Russian).
[3]. R.I. Nigmatulin. Dynamics of multiphase media . Vol .1-2, Hemisphere, New-York,1991.
[4]. L.D. Landau, E.M. Lifshitz Fluid Mechanics (Volume 6 of A Course of Theoretical Physics) Pergamon Press 1959
[5]. G. Wallis. One dimensiona Two-Phase Flow. McGraw-Hill, New York, 1969.
[6]. B.L. Kantsyrev. Riemann Problem When Modeling Dual-Speed Bubble Flow.//Proceedings of ICAPP'06, Reno, NV USA, June 4-8, 2006, Paper 6016, pp 1526-1529
[7]. B.L. Kantsyrev. Modification of Godunov Computation Method for Modeling Non-Stationary Gas- Liquid Flow.// Proceedings of ICAPP'07.Paper 7061, Nice Acropolis, France,May 13-18, 2007

Численное моделирование стратифицированных течений с использованием OpenFOAM

[1]*Н.Ф. Димитриева <dimitrieva@list.ru>*
[2]*Я.В. Загуменный <zagumennyi@gmail.com>*
[1]*ИПМехРАН, 119526, Россия,г. Москва, пр-тВернадского,д.101, корп. 1*
[2]*ИГМНАНУ, 03680, Украина, г. Киев, ул. Желябова, д.8/4,*

Аннотация. Работа посвящена построению численной модели и расчету течений непрерывно стратифицированных жидкостей в поле внешних массовых сил с учетом диссипативных факторов – эффектов вязкости и диффузии.Математическое моделирование проводится на основе фундаментальной системы дифференциальных уравнений механики неоднородных многокомпонентных жидкостей. В полной нелинейной постановке решение поставленной задачи строится численно с использованием метода конечных объемов в рамках открытого пакетаOpenFOAM. Для учета эффектов стратификации и диффузии был разработан и протестирован собственный решатель stratifiedFoam, созданный на базе стандартныхи расширенных инструментов пакета. Внимание уделяется созданию качественной расчетной сетки, удовлетворяющей требованиям разрешения всех микромасштабовзадачи в областях высоких градиентовфизических переменных. Расчеты, проведенные в параллельном режиме с использованием вычислительных ресурсов web-лаборатории UniHUB, показали высокую работоспособность предложенной численной модели и хорошее согласие с экспериментальными данными.

Ключевые слова: численное моделирование; открытые вычислительные пакеты; стратифицированные течения.

1. Введение

Исследования последних лет показали, что природные системы обладают «тонкой структурой», в которой области со сравнительно медленными изменениями параметров разделяются более тонкими границами с высокими градиентами определяющих физических величин. Под действием гравитационной силы растворенные в жидкости вещества распределяются неравномерно и формируют устойчивую стратификацию, задаваемую профилями концентрации примеси [1, 2]. Такая неравновесная среда характеризуется наличием ряда специфических движений жидкости, включая индуцированные диффузией течения на неподвижных непроницаемых

препятствиях, характеризующихся сложной ячеистой структурой течения[2–5], а также системы внутренних волн и тонкоструктурных прослоек, образующихся при движении препятствий в толщестратифицированной жидкости [6,7].

Эффекты стратификации, которые активно изучаются теоретически и экспериментально, находят разнообразные приложения в гидроаэродинамике природных и индустриальных систем. Научный интерес к данной проблеме обусловлен необходимостью изучения ряда явлений в окружающей среде, таких как интенсивные долинные или горные ветры в атмосфере и склоновые потоки в океане [2, 3], а также самодвижение объектов [8, 9]. Гравитационные (внутренние) волны являются важным элементом динамики морской среды и атмосферы, они переносят на большие расстояния энергию и импульс, интенсифицируют перенос вещества и влияют на безопасность полетов в атмосфере [6, 7].

В силу сложности такого типа задач для теоретического анализа одним из основных инструментов их решения становится численное моделирование. Современные вычислительные методы позволяют исследовать характеристики течения в полной нелинейной постановке с учетом всех диссипативных факторов и естественных переменныхбез введениядополнительных ограничений(приближенийпограничного слоя либо подстилающей плоскости, необходимых для построения аналитических решений [6]), которые затрудняют выполнение количественных сравнений с данными эксперимента [10, 11].

Целью данной работы является развитие методики численного моделирования динамики и тонкой структуры течений стратифицированных жидкостей с учетом реальных свойств среды, геометрии течений и влияния внешних динамических факторов.

2. Математическая постановка задачи

В данной работе решается нестационарная плоская задача формирования теченийнепрерывно стратифицированных жидкостей около неподвижных и движущихся непроницаемых препятствия выбранной формы.

2.1 Система уравнений

В качестве базовой математической модели для изучаемых физических процессов выбрана система дифференциальных балансных уравнений механики неоднородных многокомпонентных жидкостей в приближении Буссинеска и пренебрежении эффектами сжимаемости, поскольку скорости изучаемых течений малы по сравнению со скоростью звука [1, 10]. Она включает в себяуравнение состояния $\rho\left(S\left(y\right)\right)$, неразрывности Даламбера, баланса вещества Фикаи импульса Навье-Стокса:

$$div\mathbf{v} = 0, \qquad \rho = \rho_{00}\left(\exp\left(-y/\Lambda\right) + s\right),$$
$$\frac{\partial s}{\partial t} + \mathbf{v}\cdot\nabla s = k_s \Delta s + \frac{v_y}{\Lambda}, \qquad \frac{\partial \mathbf{v}}{\partial t} + \left(\mathbf{v}\nabla\right)\mathbf{v} = -\frac{1}{\rho_{00}}\nabla P + \nu\Delta\mathbf{v} - s\mathbf{g}, \tag{1}$$

где s – возмущение солености (стратифицирующей примеси), включающее коэффициент солевого сжатия, $\mathbf{v} = \left(v_x, v_y\right)$ – скорость жидкости, где ось $0y$ направлена вертикально вверх, P – давление за вычетом гидростатического, ν – коэффициент кинематической вязкости, k_s – коэффициент диффузии соли, t – время, $\mathbf{g}$– ускорение свободного падения, ∇ и Δ – операторы Гамильтона и Лапласа, $\Lambda = \left(d\ln\rho_0/dy\right)^{-1}$ – длина, $N = \sqrt{g/\Lambda}$ – частота и $T_b = 2\pi/N$ – период плавучести.

2.2 Начальные и граничные условия

Постановка задачи предполагает, что в начальный момент времени $t = 0$ в покоящуюся непрерывно стратифицированную жидкость помещается непроницаемое препятствие, наповерхности которого задается условие прилипания для скорости и непротекания для вещества:

$$\mathbf{v}, s\big|_{t\le 0} = 0, \quad v_{x,y}\big|_{\Sigma} = 0, \quad \mathbf{v}, s\big|_{x,y\to\infty} = 0, \quad \left.\frac{\partial S}{\partial n}\right|_{\Sigma} = -\frac{1}{\Lambda}\frac{\partial y}{\partial n} + \left.\frac{\partial s}{\partial n}\right|_{\Sigma} = 0, \tag{2}$$

где n – внешняя нормаль к поверхности препятствия $\sum$.На большом удалении от препятствия задаются условия затухания всех возмущений.

Система уравнений (1) с граничными условиями (2) описывают течения, индуцированные диффузией,которые характеризуются сложной многоуровневой системой циркуляционных движений жидкости, компенсирующих прерывание диффузионного потока стратифицирующей примеси на непроницаемом препятствии [5]. Установившееся поля физических переменных такого течения служат начальнымиусловиямидля задачи обтекания препятствия потоком непрерывно стратифицированной жидкости, когда на удалении от препятствия задается невозмущенный поток:

$$v_x\big|_{x,y\to\infty} = U, \quad v_y\big|_{x,y\to\infty} = 0.$$

2.3 Характерные масштабы задачи

Задача характеризуется набором размерных параметров ($\nu = 10^{-6}$ м^2/с, $k_s = 1.41\cdot 10^{-9}$ м^2/с, g = 9.8 м/с^2, $N = 0 \div 1.2$ с$^{-1}$), которые формируют характерные масштабы:времени $t = T_b$, длины – масштаб плавучести Λ,

препятствия L, вязкий $\delta_N^{\nu} = \sqrt{\nu/N}$ и диффузионный $\delta_N^{\kappa_S} = \sqrt{\kappa_S/N}$ микромасштабы,скорости $-U_N^{\nu} = \sqrt{\nu N}$, $U_N^{\kappa_S} = \sqrt{\kappa_S N}$.

Отношения масштабов задают традиционные безразмерные комплексы – шкалу плавучести $C = \Lambda/L$, числа Рейнольдса $\mathrm{Re} = |\mathbf{v}|L/\nu$, Пекле $\mathrm{Pe} = |\mathbf{v}|L/\kappa_S$, Шмидта $\mathrm{Sc} = \nu/\kappa_S$ и Фруда $\mathrm{Fr} = |\mathbf{v}|/NL$, где $|\mathbf{v}|$ – величина характерной скорости течения, которая при рассмотрении задачи течения, индуцированного диффузией на неподвижном препятствии, принимается равной $U_N^{\kappa_S}$, а при решении задачи об обтекании препятствия – скорости потока *U*.

Существенные различия в значениях характерных масштабов длины указывают на сложность внутренней структуры стратифицированного течения, включающей как крупномасштабные элементы – волны и вихри, так и тонкоструктурные – высокоградиентные тонкие прослойки[10].

Основное достоинство рассмотренной постановки задачи в том, что она позволяет одновременно изучать все элементы течений в рамках единого описания в естественных физических переменных без привлечения дополнительных констант и связей.

3. Численное моделирование

Изначально поставленная задача решалась методом конечных разностей на основе программ собственной разработкина языке программирования Фортран с использованием персональных компьютеров [5]. Результаты расчетов показали достаточно хорошее совпадение с аналитическими и экспериментальными данными, однако дальнейшего развития данный подход не получил ввиду необходимости использования более точных численных методик и высокого пространственно-временного разрешения, что потребовало применения высокопроизводительных вычислительных систем.

Анализ коммерческих пакетов прикладных программ с закрытым исходным кодом показал, что в них на сегодняшний день нет готовых решений системы фундаментальных уравнений многокомпонентных жидкостей. Заметный прогресс в решении сложных задач механики сплошных сред обусловлен развитием открытых вычислительных технологий, которые позволили реализовать более точные методы построения решений и высокоразрешающие численные модели. Одним из наиболее перспективных свободно распространяемых пакетов является OpenFOAMс открытым исходным кодом [12]. Пакет представляет собой набор библиотек, утилит и решателей, предоставляющих инструменты для численного моделирования широкого ряда прикладных задач с возможностью распараллеливания вычислений.

3.1 Дискретизация расчетной области

Основными критериями для оценки параметров области решения и степени ее пространственной дискретизации являются макромасштабы Λ и L, на основе которых определяются размеры расчетной области, и микромасштабы δ_N^{ν} ми $\delta_N^{\kappa_S}$ м, задающие минимальные размеры ячеек расчетной сетки.Условия адекватного разрешения тонкоструктурных компонент течения предполагают умещение на минимальном микромасштабе нескольких ячеек расчетной сетки, главным образом, вблизи границ препятствия, где фиксируются наиболее высокие значения градиентов [13].

Проведенные тестовые расчеты с различными разрешениями расчетной сетки подтвердили необходимость удовлетворения указанному критерию выбора минимального размера ячейки. При проведении расчетов на достаточно грубой сетке вблизи препятствия фиксируются локальные нефизические осцилляции решения, которые при продолжительных вычислениях приводят к накоплению погрешности и остановке счета.

Дискретизация расчетной области осуществлялась с использованием утилит blockMesh, topoSet и refineMeshв открытом пакетеOpenFOAM, а также в открытой интегрируемой платформе SALOME. Простота геометрии позволяет построить блочно-структурированную гексаэдральную расчетную сетку с совмещением линий на границах блоков. Процедура построения была параметризирована, что позволило существенно сократить время перестройки сетки при изменении геометрических параметров расчетной области и препятствия.

Алгоритм разбиения расчетной области предполагает сгущение ячеек в направлении препятствия при условии сохранения соотношения размеров граней гексаэдров не более 2. Однако в этом случае необходимость измельчения сетки в одной подобласти течения влечет излишне мелкую сетку в других областях, где особой потребности в мелкой сетке нет, что приводит кнерациональному использованию вычислительных ресурсов.

С целью улучшения качества дискретизации расчетной области дополнительно использовались утилиты topoSet и refineMesh, позволяющие на основе геометрических либо параметрических признаков выделять подобласти расчетной сетки и локально измельчать их в соответствии с заданными масштабами и выбранными направлениями. В этом случае удается построить расчетную сетку с минимальным размером ячейки $2 \cdot 10^{-5}$ м вблизи непроницаемых границ препятствия, что удовлетворительно разрешает диффузионный микромасштаб $\delta_N^{\kappa_S}$, при относительно небольшом общем количестве ячеек $4.4 \cdot 10^5$.В то же время при построении расчетной сетки первым упомянутым способом с таким же минимальным размером ячейки вблизи границ препятствия требуется более миллиона расчетных ячеек, что существенно увеличивает время расчета.

3.2 Численноерешение

Численное моделирование системы уравнений (1) с граничными условиями (2)проводится на базе пакета OpenFOAM, открытость исходного кода которого позволила построить собственный решатель stratifiedFoam, численно реализующий построенную математическую модель с использованием метода конечных объемов. Для учета эффектов стратификации и диффузии стандартный решатель icoFoam, реализующий нестационарные уравнения Навье-Стокса для случая однородной жидкости, был дополненновыми переменными(плотность и возмущение солености) и соответствующими уравнениями для их расчета, а также новыми вспомогательными параметрами(частота плавучестиN, масштаб стратификации Λ, коэффициент диффузии k_s, ускорение свободного падения $\mathbf{g}$и др.). В уравнение Навье-Стокса для вертикальной компоненты скорости добавлены члены, учитывающие наличие стратифицирующей примеси, а в уравнение диффузии для возмущения солености – дополнительные слагаемые, определяющие фоновую стратификацию.

Для интерполяции конвективных членов использовалась TVD схема (TotalVariationDiminishing) с ограничителем (limitedlineardifferencing), которая вносит минимальную численную диффузию и обеспечивает отсутствие осцилляций решения [14]. На ортогональных участках сетки нормальные градиенты скорости на поверхности ячейки, необходимые при вычислении диффузионных членов по теореме Гаусса, находились из значений скорости в центроидах соседних ячеек по схеме второго порядка. На неортогональных участках использовалась итерационная процедура коррекции погрешности, вызванной неортогональностью сетки. Для дискретизации производной по времени использовалась неявная трехточечная несимметричная схема второго порядка с разностями назад (backwarddifferencing), которая обеспечивает хорошее разрешение физического процесса во времени.

При дискретизации граничных условий (2) наряду со стандартными утилитами и библиотеками пакета OpenFOAM использовались расширенные, входящие в группyswak4Foam. Граничное условие для возмущения солености на непроницаемой поверхности препятствия реализовано с использованием библиотеки funkySetBoundaryField, которая позволяет задавать аналитические выражения для различных физических переменных в выбранных подобластях границы расчетного домена.

Для решения полученной системы линейных алгебраических уравнений применялись итерационные солверы PCG, использующие методы сопряженных градиентов с предобусловливанием для симметричных матриц, а для асимметричных матриц – метод бисопряженных градиентов PBiCG с предобусловливанием. В качестве предобусловливателя для симметричных матриц была выбрана процедура DIC, основанная на упрощенной схеме неполной факторизации Холецкого, а для асимметричных матриц использовался предобусловливатель DILU, основанный на упрощенной

неполной LU факторизации. Для связанного расчета поля скорости и давления использовался устойчивый, хорошо сходящийся алгоритм PISO ,который показывает высокую эффективность при решении нестационарных задач.

3.3 Вычисления и обработка данных

Расчеты поставленных задач проводились в параллельном режиме с использованием ресурсов виртуальной вычислительной лаборатории UniHUB (www.unihub.ru)[15]. Технологическая платформа UniHUB ориентирована на повышение эффективности процессов разработки, внедрения и моделирования вычислительных задач и предоставляет прямой доступ на вычислительный сегмент кластера МСЦ РАН.

Проведение параллельных вычислений тестовой задачи обтекания горизонтальной пластины с достаточно высоким пространственным разрешением расчетной области показали существенную эффективность распараллеливания счета. Так, при вычислениях задачи на 8 ядрах расчет одной итерации занял около 37 сек, на 16 ядрах – 7 сек, а на 24 – 3 сек. С дальнейшим увеличением числа задействованных ядер скорость вычислений практически не меняется, т.е. в данном конкретном случае проведение расчетов на 24 ядрах кластерной системы является наиболее оптимальным.

Для проведения полного анализа структуры и динамики стратифицированных течений вычислялись дополнительные физические переменные: полная плотность, завихренность, скорость диссипации механической энергии, компоненты тензора вязких напряжений, темпа бароклинной генерации завихренности, распределения динамических характеристик и др. Для этих целей использовались стандартные утилиты пакета vorticity, stressComponents, wallGradUи др.,а также расширенные библиотеки –funkySetFields и swakTopoSources, позволяющие задавать аналитические выражения для искомых величин в определенных подобластях решения. Для преобразования цифровых данных в кодахOpenFOAMк другим форматам и для последующей обработки результатов расчетов в графических пакетах ParaView и Origin использовались стандартные утилиты sample, probesLocation и topoSet.

4. Результаты и обсуждение

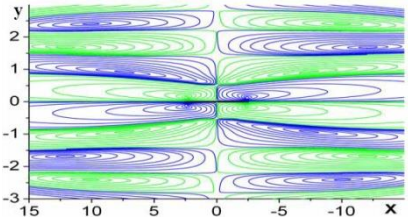
В качестве иллюстрации работоспособности разработанного решателя пакета OpenFOAM в данной работе приводятся результаты расчета обтекания горизонтальной пластины конечной толщины потоком непрерывно стратифицированной жидкости и течений, индуцированных диффузией на непроницаемой клиновидном препятствии.

4.1 Горизонтальная пластина

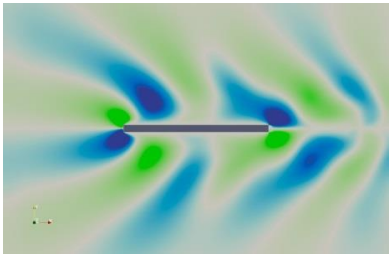
Непроницаемое препятствие, погруженное в толщу непрерывно стратифицированной среды, прерывает диффузионный поток

стратифицирующего вещества и формирует сложную структуру компенсационных движений жидкости. Картина течения, индуцированного диффузией на горизонтальной пластине, состоит из многоуровневой последовательности симметрично расположенных циркуляционных ячеек(рис.1а), влияние которых распространяется далеко за пределы препятствия в виде протяженных слоистых структур[5].

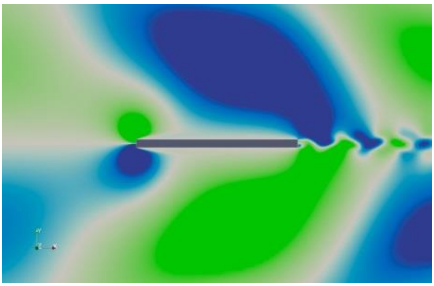
С уменьшением величины стратификации среды наблюдается уширение течений наряду с падением интенсивности циркуляции жидкости в вихревых ячейках. В случае предельно слабых стратификаций скорость циркуляционного движения жидкости оказывается чрезвычайно малой, а в приближении однородной среды такой вид течений и вовсе отсутствует. Сравнения рассчитанных полей течения, индуцированного диффузией на непроницаемой горизонтальной пластине, с картинами теневой визуализации в лабораторных опытах показали хорошее согласие численных и экспериментальных данных [10, 11].



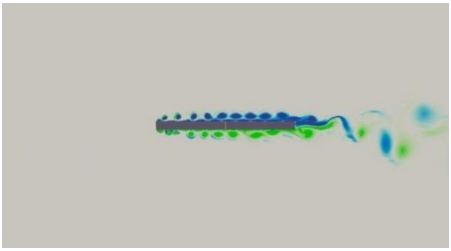
а)



б)



в)



г)

Рис.1.Структура стратифицированного течения($N=1.2c^{-1}$) около неподвижной (а – течение, индуцированное диффузией) и равномерно движущейся(б – $Re=10^3$ в $Re=5\cdot10^3$, г – $Re=8\cdot10^4$) горизонтальных пластин(L = 10 см, h = 0.5 см).

С началом обтекания препятствия потоком стратифицированной жидкости картина течения кардинально меняется: начинают формироваться опережающие возмущения, поля присоединенных внутренних волн и спутный

след [6, 7, 16]. Различаются три основных режима течения стратифицированной жидкости около горизонтальной пластины при, соответственно, малых ($Re<10^3$), умеренных ($Re<2\cdot10^4$) и сравнительно больших ($Re<10^5$)числах Рейнольдса, когда наиболее проявлены определенные структурные элементы течения: ярко выраженные опережающие возмущения и присоединенные внутренние волны (рис.1б);интенсивные внутренние волны, сосуществующие с зарождающимся спутным следом за препятствием (рис.1в);доминирующие вихревые структуры, порождаемые передними острыми кромками пластины, и интенсивный спутный вихревой след (рис.1г).

Сравнения рассчитанных картин стратифицированного течения около горизонтальной пластины дает хорошее согласие с экспериментальными теневыми картинами, в которых отчетливо просматриваются все основные структурные элементы течений [7, 10, 11].Рассмотрение различных видов жидкостей (сильно и слабо стратифицированные, потенциально и актуально однородные) при численном решении задачи обтекания препятствий обеспечивает дополнительные возможности независимого контроля точности вычислений наряду с традиционными, обеспечивающими дискретное выполнение законов сохранения с наперед заданной точностью.

4.2 Клиновидное препятствие

Большой интерес представляет изучение течений не только на полосе, но и препятствиях другой формы, особенно клиновидной. Экспериментальные исследования показали, что свободное клиновидное препятствие нейтральной плавучести, погруженное в непрерывно стратифицированную жидкость, совершает самодвижение со скоростью порядка сантиметра в час [8, 9].

Непроницаемое клиновидное препятствие блокирует фоновый диффузионный перенос и формирует сложную систему течений. Численное моделирование течений, индуцированных диффузией на клине, показало существование сложной многоуровневой системы циркуляционных ячеек, примыкающих к острым кромкам препятствия. В поле завихренности тонкий слой циклонической завихренности (против часовой стрелки), примыкает непосредственно к нижней грани клина (рис. 2). За ним следует совокупность чередующихся компенсационных областей с различными знаками. При этом интенсивность завихренности уменьшается в направлении от препятствия, а толщина слоя, наоборот, увеличивается. Распределение значений завихренности в верхней полуплоскости антисимметрично относительно горизонта нейтральной плавучести.

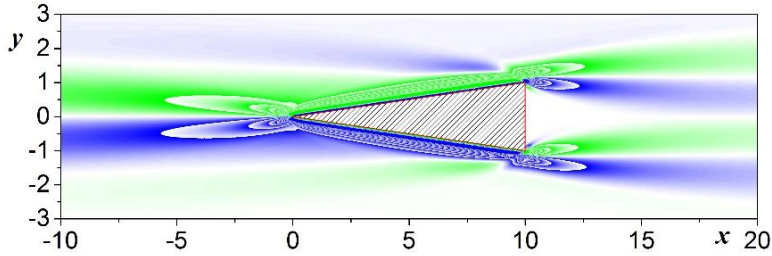


Рис. 2. Поле завихренности, индуцированного диффузией на клине длиной L = 10 см и высотой основания h = 2 см, N = 1 $с^{-1}$.

Численные исследования показали, что сложная ячеистая структура течений, формирующихся около клина, сопровождается возникновением областей дефицита давления у его острой вершины, что объясняет возникновение пропульсивной силы, приводящей к самодвижению вдоль горизонта нейтральной плавучести. С целью детального изучения влияния формы препятствия на эффект самодвижения рассмотрены клинья с прямыми и искривленными гранями симметрично относительно продольной оси *x*. Научный и практический интерес представляет влияние на динамику и структуру стратифицированных течений радиуса и знака кривизны клиновидного препятствия (выпуклость и вогнутость).

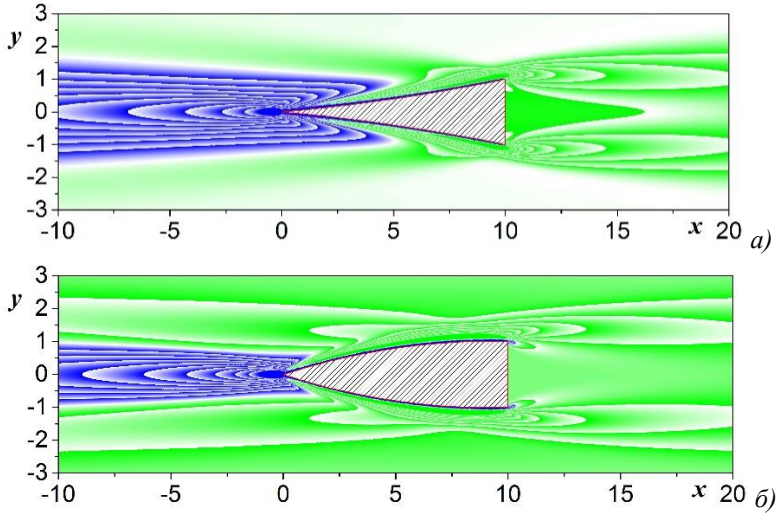


Рис. 3. Поле давления для клиновидного препятствия длиной L = 10 см и высотой основания h = 2 см с искривленными гранями: вогнутыми (а) и выпуклыми (б).

Результаты вычислений поля давления, представленные на рис. 3, показали, что размеры области дефицита давления у острой вершины клина зависят от

формы искривления боковых сторон. Зона отрицательного давления захватывает часть поверхности вблизи острой вершины клиновидного препятствия с вогнутыми гранями (рис. 3а) и простирается в горизонтальном направлении. При этом толщина этой зоны значительно больше, чем для выпуклого клина (рис. 3б).

5. Заключение

Предложена численная модель расчета течений непрерывно стратифицированной жидкости с учетом эффектов нелинейности и диффузии, реализованная методом конечных объемов в решателе собственной разработки открытого пакета OpenFOAM.

Отработана процедура построения качественной расчетной сетки, удовлетворяющей требованиям разрешения всех микромасштабов задачи в высокоградиентных областях течения.

Расчеты стратифицированных течений около пластины и клина, проведенные в параллельном режиме с использованием вычислительных ресурсов web-лаборатории UniHUB, показали высокую работоспособность предложенной численной модели и хорошее согласие с экспериментальными данными**.**

Благодарности

Работа выполнена при финансовой поддержке РФФИ (проект 14-37-50001). Авторы выражают глубокую благодарность зав. Лабораторией механики жидкостей ИПМех РАН, проф. Ю.Д. Чашечкину за конструктивные идеи и поддержку, а также сотрудникам ИСП РАН, предоставившим вычислительные ресурсы web-лаборатории UniHUB.

Список литературы

[1]. Л.Д.Ландау, Е.М.Лифшиц. Теоретическая физика, том VI, Гидродинамика. М.: Наука, 1986. 736 с.

[2]. Л. Прандтль.Гидроаэромеханика. М.: ИИЛ, 1949 г. 488 с.

[3]. O.M.Phillips. On flows induced by diffusion in a stably stratified fluid. Deep-Sea Res.,volume 17, 1970. P. 435–443.

[4]. A. Shapiro, E. Fedorovich. A boundary-layer scaling for turbulent katabatic flow. Boundary-layer meteorology, volume 153, Issue 1, 2014. P. 1-17. doi: 10.1007/s10546-014-9933-3.

[5]. Я.В. Загуменный, Ю.Д. Чашечкин. Тонкая структура нестационарного течения, индуцированного диффузией на неподвижной пластине. Известия РАН: Механика жидкости и газа, № 3, 2013 г.стр. 100-117.

[6]. Ю.Д.Чашечкин, Р.Н.Бардаков, Я.В.Загуменный. Расчет и визуализация тонкой структуры полей двумерных присоединенных внутренних волн, Морской гидрофизический журнал, № 6. 2010 г. стр. 3-15.

[7]. Я.В.Загуменный. Динамика и структура стратифицированного течения около горизонтальной пластины. Доклады НАН Украины, № 7, 2014 г. стр. 60-67.

[8]. M. R.Allshouse, M.F.Barad, T.Peacock Propulsion generated by diffusion-driven flow. Nature Physics,volume 6, 2010. P. 516–519. doi: 10.1038/nphys1686

[9]. M.J.Mercier, F.M.Ardekani, M.R.Allshouse, B.Doyle, T.Peacock. Self-propulsion of immersed object via natural convection. Physical review letters, volume 112, 2014. P. 204501(5).

[10]. Ю.Д.Чашечкин. Дифференциальная механика жидкостей: наблюдения и расчеты структуры течений. Журнал проблем эволюции открытых систем, том 2, вып. 15, 2013 г. с. 20-36.

[11]. Yu.D.Chashechkin, V.V.Mitkin.A visual study on flow pattern around the strip moving uniformly in a continuously stratified fluid, J. Visualiz, volume 7,Issue 2, 2004. P. 127-134.

[12]. В.Т.Калугин, М.В.Крапошин, С.В.Стрижак, А.В.Юскин Возможности открытого пакета OpenFOAM для решения задач аэрогидромеханики и теплообмена. Труды РНКТ-5. М.: Изд. дом МЭИ, том 1, 2010 г. с. 85-88.ISBN 978-5-383-00529-3

[13]. Ю.Д.Чашечкин, Я.В.Загуменный. Расчет течений непрерывно стратифицированной жидкости с использованием открытых вычислительных пакетов на базе технологической платформы UniHUB. Труды Института системного программирования РАН, том 24, 2013 г. стр. 87-106. doi: 10.15514/ISPRAS-2013-24-5.

[14]. Д.В.Чирков, С.Г.Черный. Сравнение точности и сходимости некоторых TVD-схем. Вычислительные технологии, том 5, 2000 г. с. 86-107.

[15]. О. Самоваров, С. Гайсарян. Архитектура и особенности реализации платформы UniHUB в модели облачных вычислений на базе открытого пакета OpenStack. Труды Института системного программирования РАН, том 26, вып. 1, 2014 г. стр. 403-420 doi: 10.15514/ISPRAS-2014-26(1)-17.

[16]. N.F.Dimitrieva, Ia.V. Zagumennyi. Calculations of admixture transport around a horizontal plate in a continuously stratified fluid. Selected papers of international conference “Fluxes and structures in fluids”. M.: MAKS Press, 2014. P. 61–68.

Numerical simulation of stratified flows usingOpenFOAM package

[1]*N.F. Dimitrieva <dimitrieva@list.ru>*
[2]*Ya.V. Zagumennyi < zagumennyi@gmail.com>*
[2]*IHMNASU, 03680, Ukraine, Kiev, 8/4Zheliabova Street,*
[1]*IPMech RAS, 119526, Russia, Moscow, 101/1 Vernadskogo Avenue.*

Abstract. The paper is devoted to construction of a numerical model and computations of continuously stratified fluid flows in field of external mass forces accounting for dissipative factors, viscosity and diffusion. Mathematical model is based on the fundamental set of differential equations of inhomogeneous multicomponent fluid mechanics. Solution of the problem is constructed numerically in the completenon-linear formulation using finite volume method in frame of the open source package OpenFOAM. To take in to account the stratification and diffusion effects a new own solver, stratifiedFoam, was developed and tested using the standard and extended libraries of the package. A particular attention is focused at construction of a high quality computational grid which satisfies basic requirements for resolution of all the microscales of the problem in high-gradient regions of the flow. The calculations performed in parallel regime on computational facilities of the web-laboratory UniHUB demonstrated a pretty high efficiency of the proposed numerical model and a good agreement with the experimental data.

Keywords: numerical simulation; open source computational packages; stratified flows.

References

[1]. L.D. Landau, E.M. Lifshits. Teoreticheskaya fizika, tom VI Gidrodinamika [Theoretical physics, volume VI, Hydrodynamics]. M.:Nauka [Moscow: Science], 1986. 736 p.(in Russian)
[2]. L.Prandtl. Gidroaehromekhanika [Hydroaeromechanics]. M.: Izdatel′stvo inostrannoi literatury [Moscow: Foreign Literature Publishing House], 1949. 488 p.(in Russian)
[3]. O.M.Phillips. On flows induced by diffusion in a stably stratified fluid. Deep-Sea Res.,volume 17, 1970. P. 435–443.
[4]. A. Shapiro, E. Fedorovich. A boundary-layer scaling for turbulent katabatic flow. Boundary-layer meteorology, volume 153, Issue 1, 2014. P. 1-17. doi: 10.1007/s10546-014-9933-3.

[5]. Ia.V.Zagumennyi, Yu.D.Chashechkin. Fine structure of unsteady diffusion-induced flow over a fixed plate. Fluid Dynamics, volume 48, № 3, 2013.P. 374-388. doi: 10.1134/S0015462813030113

[6]. Yu.D.Chashechkin, R.N. Bardakov, Ia. V. Zagumennyi. Raschet i vizualizatsiya tonkoj struktury polej dvumernykh prisoedinennykh vnutrennikh voln [Calculation and visualization of the fine structure of fields of two-dimensional attached internal waves], Morskoj gidrofizicheskij zhurnal [MarineHydrophysicalJournal], № 6. 2010. P. 3-15 (inRussian).

[7]. Ia. V. Zagumennyi.Dinamika i struktura stratifitsirovannogo techeniya okologorizontal'noj plastiny [Dynamics and structure ofa stratifiedflow arounda horizontalplate]. Doklady NAN Ukrainy [Reports of NAS of Ukraine], № 7, 2014. P. 60-67(in Russian).

[8]. M. R.Allshouse, M.F.Barad, T. Peacock Propulsion generated by diffusion-driven flow. Nature Physics, volume 6, 2010. P. 516–519. doi: 10.1038/nphys1686

[9]. M.J.Mercier, F.M.Ardekani, M.R.Allshouse, B.Doyle, T.Peacock. Self-propulsion of immersed object via natural convection. Physical review letters, volume 112, 2014. P. 204501(5).

[10]. Yu.D.Chashechkin. Differentsial'naya mekhanika zhidkostej: nablyudeniya i raschety struktury techenij [Differential fluid mechanics: flow, structures observations and calculations] ZHurnal problem ehvolyutsii otkrytykh sistem [Journal of the problems of the evolution of open systems], volume 2, Issue. 15, 2013. P. 20-36 (in Russian).

[11]. Yu.D.Chashechkin, V.V.Mitkin. A visual study on flow pattern around the strip moving uniformly in a continuously stratified fluid, J. Visualiz, volume 7, Issue 2, 2004. P. 127-134.

[12]. V. T. Kalugin, M. V. Kraposhin, S. V. Strizhak, A. V.YuskinVozmozhnosti otkrytogo paketa OpenFOAM dlya resheniya zadach aehro gidro mekhanikii teploobmena [The possibility of the opening package OpenFOAM to solve aero hydro mechanic and heat transfer problems]. Trudy RNKT-5 [Proceedings of the Fifth Russian National Conferenceon Heat Transfer]. M.:Izdatel'skij dom MEHI [Moscow: MPEI-publisher], volume 1, 2010. P. 85-88 (in Russian). ISBN 978-5-383-00529-3

[13]. Yu.D.Chashechkin, Ia. V. Zagumennyi. Raschet techenij nepreryvno stratifitsirovannoj zhidkosti s ispol'zovaniem otkrytykh vychislitel'nykh paketov na baze tekhnologicheskoj platformy UniHUB [Calculation of continuously stratified fluid flows using open computational packages based on the technological platform UniHUB]. Trudy ISP RAN [The Proceedings of ISP RAS], volume 24, 2013. P. 87-106(in Russian).doi: 10.15514/ISPRAS-2013-24-5.

[14]. D.V.Chirkov, S.G.Chernyi. Sravnenie tochnosti i skhodimosti nekotorykh TVD-skhem [Comparison of accuracy and convergence rate of someTVD-schemes] Vychislitel'nye tekhnologii [ComputerApplications], volume 5, 2000. P. 86-107 (in Russian).

[15]. O. Samovarov, S. Gaysaryan. Arkhitektura i osobennosti realizatsii platformy UniHUB v modeli oblachnykh vychislenij na baze otkrytogo paketa OpenStack [The web-laboratory architecture based on the cloud and the UniHUB implementation as an extension of the OpenStack platform]. Trudy ISP RAN [The Proceedings of ISP RAS], volume 26, issue 1, 2014. P. 403-420(in Russian) doi: 10.15514/ISPRAS-2014-26(1)-17.

[16]. N.F. Dimitrieva, Ia.V. Zagumennyi. Calculations of admixture transport around a horizontal plate in a continuously stratified fluid. Selected papers of international conference “Fluxes and structures in fluids”. M.: MAKS Press, 2014. P. 61–68.

Исследование влияния длины улиц на течение воздуха в них

[1,2] *М.В. Волик <volikmv@mail.ru>*
[1] *Финансовый университет при Правительстве РФ,*
362002, Россия, г. Владикавказ, ул. Молодежная, 7
[2] *Южный математический институт ВНЦ РАН и РСО-Алания,*
362027, Россия, г. Владикавказ, ул. Маркуса, д. 22

Аннотация. В работе проводится сравнение результатов математического моделирования аэродинамики типичных городских застроек с разной длиной улиц. Расчеты проводились с помощью свободно распространяемого пакета OpenFoam и удаленного доступа к консоли на управляющем узле вычислительного кластера BL2x220 Cluster Console https://unihub.ru/resources/bl2x220cc Web-лаборатории Unihub (www.unihub.ru) по программе «Университетский кластер» (www.unicluster.ru). Рассмотрены одиночная улица с домами одинаковой высоты по ее сторонам, одиночная улица с домами разной высоты по ее сторонам и две параллельные улицы, расположенные на склоне холма. Сравнение результатов расчетов в двумерном и трехмерном приближении показало, что длина улиц оказывает значительное влияние на качественную и количественную картины течения внутри улиц и над застройкой.

Ключевые слова: математическое моделирование; аэродинамика; городская застройка; OpenFoam.

1. Введение

В настоящее время математическое моделирование аэродинамики улиц позволяет с минимальными затратами выявлять и решать задачи экологического характера. Перспективным является исследование распространения загрязняющих веществ, выбрасываемых автомобилями, число которых катастрофически увеличивается на улицах городов, а также стационарных или аварийных выбросов производственными предприятиями. Величина концентрации загрязняющих веществ зависит не только от плотности движения автотранспорта, но и от характеристик воздушного потока в городской застройке. Прежде чем анализировать распределение загрязняющих веществ, необходимо получить картину течения воздуха внутри улиц и над ними, которая позволит определить наиболее опасные области, в которых скапливаются загрязняющие вещества.

Снижение затрат по решению поставленной задачи позволяет использование и развитие свободного программного обеспечения: открытых пакетов, подобных OpenFoam (Open Field Operation and Manipulation), Dolphyn, Salome, Engrid, Paraview и других [1].

В данной работе математическое моделирование течения воздуха в городской застройке проводилось с помощью свободно распространяемого пакета OpenFoam, который работает под различными версиями открытой операционной системы Linux (OpenSuse, Centos, Ubuntu, Fedora, Debian и другие), и удаленного доступа к суперкомпьютеру Web-лаборатории UniHUB (www.unihub.ru) по программе «Университетский кластер» (www.unicluster.ru).

2. Постановка задачи

Структура OpenFoam является полностью модульной, каждый этап численного решения базовых уравнений выносится в отдельный модуль: дискретизация расчётной области (создание сетки), дискретизация уравнений по времени и пространству, методы решения систем линейных алгебраических уравнений, граничные условия (в том числе пристеночные функции), модели турбулентности (Reynolds-Averaged Stresses, Large Eddy Simulation), контроль качества сетки [1].

В качестве постпроцессора для визуализации результатов расчетов предназначен пакет ParaView, интегрированный в OpenFoam и UniHUB. ParaView позволяет построить линии тока, графики по одной или нескольким величинам, рассчитать средние значения по объему или поверхности, рассчитать перепад давления, экспортировать данные в файлы и т.д.

Решаемая в OpenFoam задача обязательно содержит: начальные и граничные условия; расчетную сетку, а также физические свойства и параметры интегрирования уравнений [1]. Для проведения вычислительных экспериментов использовался стандартный решатель pimpleFoam для турбулентного течения жидкости, в котором применяется алгоритм связи скорости и давления Pimple. Предполагалось, что движущийся воздух является несжимаемой жидкостью. Система уравнений включала уравнение неразрывности

$$\nabla \cdot \vec{\mathbf{U}} = 0$$

и уравнение изменения импульса

$$\frac{\partial \vec{\mathbf{U}}}{\partial t} + \vec{\mathbf{U}} \cdot \nabla \vec{\mathbf{U}} - \nabla \cdot \left(\left(\nu + \nu_t \right) \nabla \vec{\mathbf{U}} \right) = -\frac{1}{\rho} \nabla P \text{, где}$$

U – вектор скорости (м/с),

t – время (с),

P – давление ($м^2/с^2$),

ρ – плотность воздуха (1.2 $кг/м^3$),

ν – ламинарная вязкость ($м^2/с$),

ν_t – турбулентная вязкость ($м^2/с$).

Турбулентность моделировалась с использованием стандартной $K-\varepsilon$ модели, для которой решались уравнения для кинетической энергии турбулентности

$$\frac{\partial}{\partial t}K+\frac{\partial}{\partial x_i}(KU_i)=\frac{\partial}{\partial x_j}\left[\left(\nu+\frac{\nu_t}{\sigma_k}\right)\frac{\partial K}{\partial x_j}\right]+P_k-\varepsilon$$

и скорости ее диссипации

$$\frac{\partial}{\partial t}\varepsilon+\frac{\partial}{\partial x_i}(\varepsilon U_i)=\frac{\partial}{\partial x_j}\left[\left(\nu+\frac{\nu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right]+C_{1\varepsilon}\frac{\varepsilon}{K}P_k-C_{2\varepsilon}\frac{\varepsilon^2}{K}$$ где

,

K – энергия турбулентности ($м^2/с^2$),

ε – скорость диссипации энергии турбулентности ($м^2/с^3$),

P_k – скорость порождения энергии турбулентности.

Турбулентная вязкость определяется как: $\nu_t=C_\mu\frac{K^2}{\varepsilon}$

Начальные значения для K и ε выбираются на основе следующих формул:

$$K=\frac{1}{2}\left(U'^2_x+U'^2_y+U'^2_z\right),\quad \varepsilon=\frac{C_\mu^{0.75}K^{3/2}}{l},$$

где $U_x'^2, U_y'^2, U_z'^2$ – усредненные квадраты пульсационных компонент скорости в направлениях x, y и z соответственно. Предполагается, что начальная турбулентность изотропна, т. е. $U_x'=U_y'=U_z'$ и равна 5% от скорости воздуха на входе, а длинна l равна расстоянию от нижней границы расчетной области до верхней. Тогда:

$$K=0.375(м^2/с^2)\quad \varepsilon=0{,}001(м^2/с^3).$$ [2]

,

В расчетах использованы следующие безразмерные константы [2] $C_\mu=0.09, \kappa=0.41, E=9.8, C_{1\varepsilon}=1.44, C_{3\varepsilon}=0.09, C_{2\varepsilon}=1.92.$

Использовались следующие граничные условия:

- для скорости: на входе фиксированное значение (10 м/с); на верхней и выходной границах – нулевой градиент ($\frac{\partial U}{\partial n}=0$); на стенках домов и дне уличного каньона – фиксированное значение, равное нулю;

- для избыточного давления: на входе, верхней и нижней границах принимался нулевой градиент ($\frac{\partial P}{\partial n} = 0$), а на выходе — фиксированное значение, равное нулю;
- для турбулентной вязкости: на входе, выходе и верхней границе – значения получены по $K - \varepsilon$ модели, на нижней границе (стенках) – использовалась пристеночная функция nutkWallFunction [2].
- для энергии турбулентности: на входе – фиксированное значение K=0.375, на выходе и верхней границе – нулевой градиент ($\frac{\partial K}{\partial n} = 0$), на нижней границе – пристеночная функция kqRWallFunction [2].
- для скорости диссипации энергии турбулентности: на входе – фиксированное значение $\varepsilon = 0.001$, на выходе и верхней границе — нулевой градиент ($\frac{\partial \varepsilon}{\partial n} = 0$), на нижней границе – пристеночная функция epsilonWallFunction [2].

Серия вычислительных экспериментов проводилась для интервала времени от 0 до 100 с. (с шагом 0.001 с.). Использовалась равномерная расчетная сетка в прямоугольной области с шагом по пространству 1 м. В качестве масштаба скорости U1 выбрано значение скорости воздуха в центре улиц на уровне крыш. Входная граница располагается слева, т.е. моделируется течение воздуха поперек улиц. Расстояние от входной границы до наветренной стороны застройки принималось равным десяти высотам, от подветренной стороны застройки до выходной границы – двадцати высотам, от нижней границы расчетной области до верхней границы – шести высотам. Исследуются три варианта конфигурации городской застройки, каждый из которых рассмотрим подробнее.

3. Моделирование течения воздуха в одиночной улице с домами одинаковой высоты по ее сторонам и сравнение с экспериментальными данными

Типичная одиночная улица с домами одинаковой высоты по ее сторонам представляет особый интерес, т.к. в разной литературе приводятся результаты экспериментов в аэродинамической трубе именно для такой конфигурации.

В качестве масштаба длины h выбрана высота домов, равная 15м. В экспериментах [3, 4] и расчетах ширина улицы принималась равной одной высоте домов. Расчет проводился в двумерном приближении, когда длина улицы составляла 1м, и трехмерном приближении при длине улицы, равном 20м.

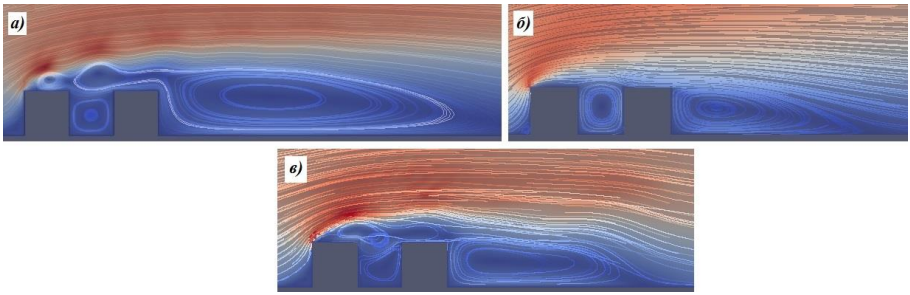


Рис. 1. Линии тока в улице с домами одинаковой высоты по ее сторонам (а – длина улицы 1м, б – длина улицы 20м, в – длина улицы 60м)

Результаты расчетов показали, что в случае двумерного варианта (рис. 1а) один вихрь образуется внутри улицы и в следе за застройкой и многовихревая структура – над всей застройкой. Внутри улицы поток воздуха перемещается против часовой стрелки, а над застройкой и в следе – по часовой стрелке. В случае трехмерного варианта (рис. 1б) получены качественно и количественно другие результаты расчетов: вихри образуются внутри улицы, в следе за застройкой и над домом на подветренной стороне. Во всех этих вихрях воздух перемещается по часовой стрелке. Кроме того, размер вихря в следе за застройкой в трехмерном случае составляет 65м, а в двумерном – 130м.

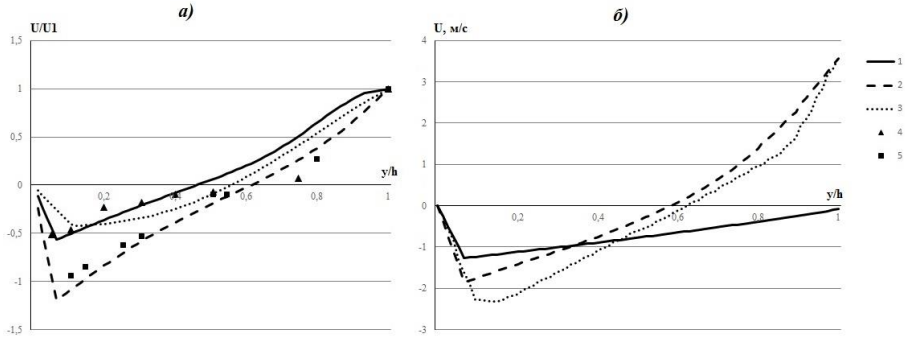


Рис.2. Распределение горизонтальной составляющей скорости воздуха по высоте в центре улиц (а) с домами одинаковой высота и в следе за застройкой (б). Кривые 1 – длина улицы 1м, кривые 2 – длина улицы 20м, кривые 3 - длина улицы 60 и/или 120м, значки 4 и 5 – экспериментальные данные [3] и [4] соответственно.

На рис.2 представлено распределение горизонтальной скорости воздуха по высоте в центре улиц и в следе за застройкой на расстоянии, равном 0.5 высоты домов. Видно, что внутри улицы центр вихря в двумерном случае

располагается ниже, чем в трехмерном, а скорость возвратного течения значительно ниже, чем при трехмерных расчетах. В следе за застройкой в двумерном случае горизонтальный размер вихря в два раза больше, чем в трехмерном, поэтому скорость потока в этом случае ниже, особенно в верхней части вихря.

Для сравнения на рис. 2а представлены экспериментальные данные. Показано, что результаты двумерных расчетов в нижней половине улицы удовлетворительно совпадают с экспериментальными данными [3]. В связи с тем, что исследование распространения загрязняющих веществ чаще всего проводится на уровне пешеходов, т.е. на высоте 2м над проезжей частью. Полученное совпадение расчетных данных с экспериментальными подтверждает возможность использования двумерных расчетов для дальнейших исследований. Кроме того, на графике показано удовлетворительное совпадение с экспериментальными данными [4] результатов трехмерных расчетов для улицы длиной 20м.

Результаты расчетов для вариантов, когда расстояние вдоль улиц составляет пять и десять высот домов, полностью совпадают между собой. Картина течения при этом (рис. 1в) качественно похожа на ту, которая получена при двумерном варианте: вихри образуются не только внутри улиц и в следе, но и над всей застройкой. Количественно же результаты несколько отличаются от двумерного варианта: горизонтальный размер вихря в следе за застройкой составляет порядка 53м, центр вихря внутри улицы располагается немного выше, а скорость возвратного течения воздуха в нем ниже (кривая 3 на рис. 2а). При длине улицы 60м и 120м изменение скорости по высоте вновь становится близким к двумерному случаю.

Таким образом, можно сделать вывод, что при проведении расчетов изменение длины улиц оказывает значительное влияние на качественные и количественные результаты для одиночной улицы с домами одинаковой высоты по ее сторонам. Это, возможно, объясняет отличия в результатах экспериментов в аэродинамических трубах.

4. Моделирование течения воздуха в одиночной улице с домами разной высоты по ее сторонам

В этом случае высота домов на наветренной стороне улицы составляла 0.5 от высоты домов на подветренной стороне, ширина улицы – одну высоту домов на подветренной стороне.

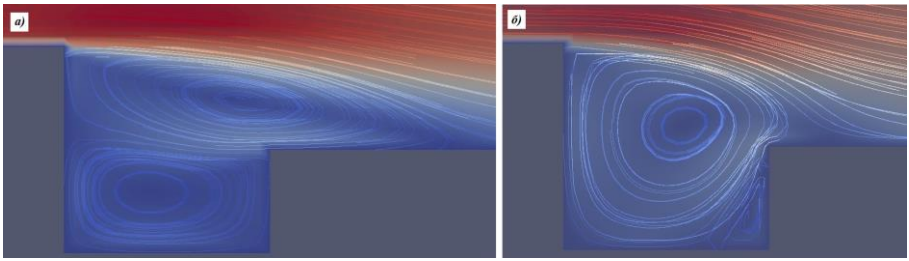


Рис. 3. Линии тока в улице с домами разной высоты по ее сторонам (а – длина улицы 1м, б – длина улицы 20м)

Результаты расчетов в двумерном приближении (рис. 3а) показали образование двухвихревой структуры. В нижнем вихре, находящимся внутри улицы, воздух вращается против часовой стрелки, а в верхнем – по часовой стрелке и затекает на крышу дома на наветренной стороне на расстояние около 20м. Скорость воздуха в нижнем вихре ниже, чем в верхнем (рис. 4).

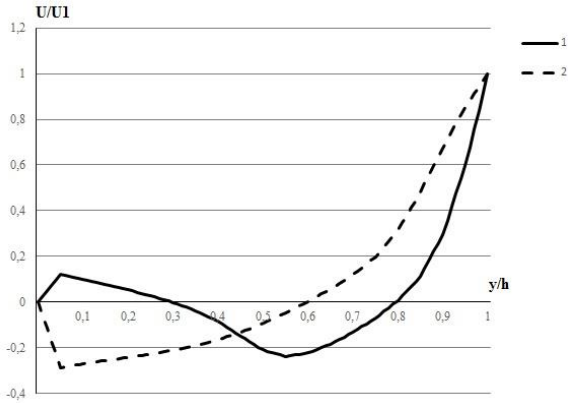


Рис.4. Распределение горизонтальной составляющей скорости воздуха по высоте в центре улицы с домами разной высоты по ее сторонам. Кривые 1 – длина улицы 1м, кривые 2 – длина улицы 20м.

С увеличением длины улицы до 20м картина течения воздуха существенно изменяется (рис. 3б). Внутри улицы, помимо основного вихря, возникает вторичный вихрь размером 4м×7м вблизи дома на наветренной стороне улицы. В основном вихре воздух перемещается по часовой стрелке, во вторичном – против часовой стрелки. Скорость воздуха в основном вихре в трехмерном случае выше, чем в двумерном.

Таким образом, как и в предыдущем варианте расчетов, изменение длины одиночной улицы с разной высотой домов оказывает значительное влияние на результаты.

5. Моделирование течения воздуха в двух параллельных улицах, расположенных на склоне холма

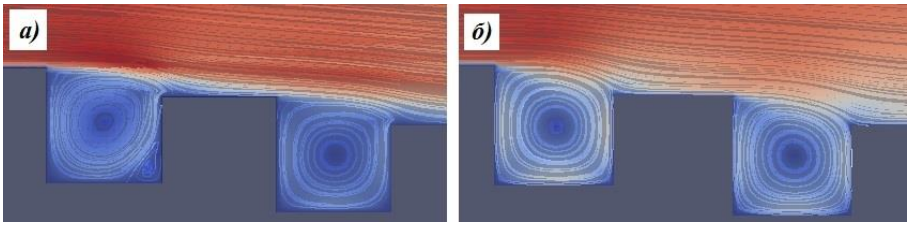


Рис. 5. Линии тока в улицах на склоне холма (а – длина улиц 1м, б – длина улиц 20м)

Разные результаты расчетов получены также при моделировании аэродинамики двух параллельных улиц, расположенных на склоне холма. Высота домов на наветренной стороне улиц в этом случае составляла 0.75 от высоты домов на подветренной стороне.

Если ширина улиц принималась равной высоте домов на подветренной стороне, картина течения в первой по потоку улице заметно отличается. В случае двумерных расчетов (рис. 5а) внутри улицы, помимо основного вихря, образуется вторичный вихрь размером 6м×8м вблизи проезжей части на наветренной стороне улицы, а в случае трехмерных расчетов (рис. 5б) вторичного вихря нет и скорость возвратного течения в основном вихре значительно выше, чем в двумерных расчетах (рис. 6а). В нижней по потоку улице в обоих вариантах образуется один вихрь, а скорость возвратного течения также больше в случае трехмерных расчетов (рис. 6б).

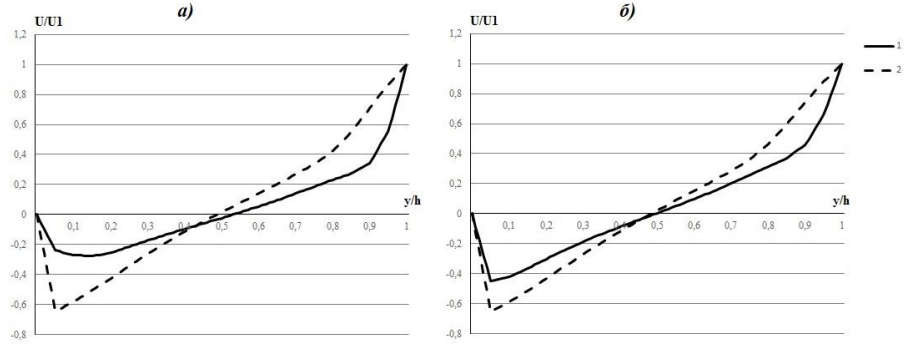


Рис.6. Распределение горизонтальной составляющей скорости воздуха по высоте в центре улиц, расположенных на склоне холма: а) улица выше по течению воздуха, б) – ниже по течению. Кривые 1 – длина улиц 1м, кривые 2 – длина улиц 20м.

Если ширина улиц, расположенных на склоне холма, принималась равной 0.5 от высоты домов на подветренной стороне, то в двумерных расчетах внутри верхней по потоку улицы образуются два вихря, расположенные друг под

другом (рис. 7а). Воздух в нижнем вихре перемещается против часовой стрелки, а в верхнем, более плоском, перемещается по часовой стрелке. Верхний вихрь заходит на крышу домов на наветренной стороне на 15м. В нижней по потоку улице образуется три вихря, расположенных друг под другом. Воздух в верхнем и нижнем вихрях перемещаются по часовой стрелке, а в вихре, расположенном между ними – против часовой стрелки. Верхний вихрь заходит на крышу дома на наветренной стороне на расстояние 12.5м.

Если длина улиц составляет 20м (рис. 7б), то в улице, расположенной выше по течению образуется также два вихря. Однако, верхний вихрь имеет значительно большие вертикальные размеры и заходит на крышу дома на наветренной стороне на расстояние около 12.5м. В нижней по потоку улице образуется только один вихрь, воздух в котором перемещается по часовой стрелке и затекает на крышу дома на наветренной стороне на 5м.

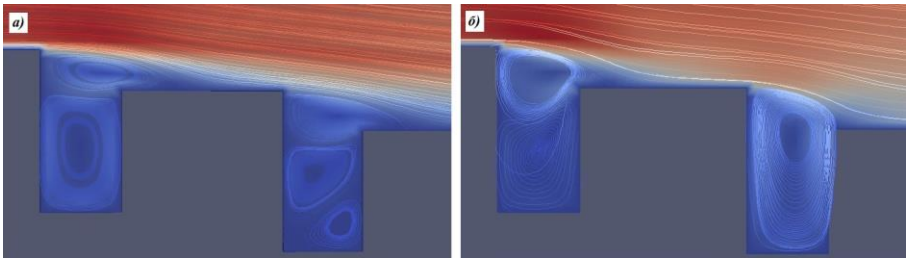


Рис. 7. Линии тока в узких улицах, расположенных на склоне холма (а – длина улиц 1м, б – длина улиц 20м)

Горизонтальная скорость потоков воздуха в центре улицы, расположенной выше по течению (рис. 8а), в случае двумерных и трехмерных расчетов почти совпадает. А в улице, расположенной ниже по течению, в верхнем вихре в случае двумерных расчетов скорость воздуха больше (рис. 8б).

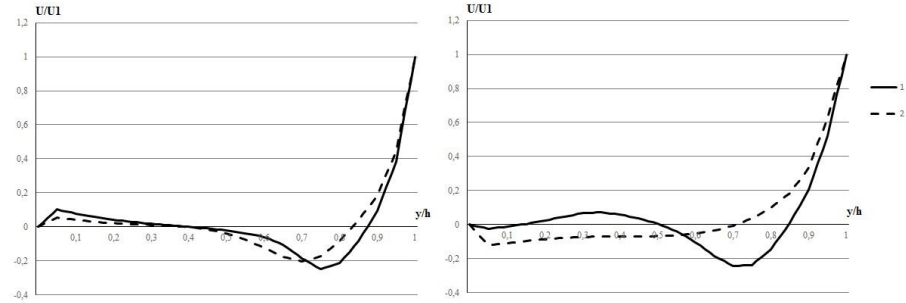


Рис.8. Распределение горизонтальной составляющей скорости воздуха по высоте в центре узких улиц, расположенных на склоне холма: а) улица выше по течению воздуха, б) – ниже по течению. Кривые 1 – длина улиц 1м, кривые 2 – длина улиц 20м.

5. Выводы

В работе показано качественное и количественное отличие вихревых структур, полученных в двумерных и трехмерных расчетах движения воздуха в одиночной улице с домами одинаковой высоты по ее сторонам, в одиночной улице с разновысокими домами и в двух параллельных улицах, расположенных на склоне холма. Причиной этого, скорее всего, является возникновение слабого течения воздуха вдоль улицы. Полученные результаты позволяют объяснить существенные отличия в экспериментальных данных различных авторов.

Список литературы

[1]. Крапошин М.В., Самоваров О.И., Стрижак С.В. Пакет OpenFoam: численное моделирование задач МСС // Материалы школы-семинара «Основы использования OpenFoam, Salome, ParaView», https://unihub.ru/tools/unicfdc1/svn/trunk/Version2/Pdf/day1_2_4-OpenFOAM-Base.pdf.

[2]. Каменецкий Е.С., Волик М.В., Тагиров А.М. Математическое моделирование распространения загрязняющих веществ, выбрасываемых автотранспортом // Известия Кабардино-Балкарского научного центра РАН, № 6(62), 2014 г. – с.23-32.

[3]. Uehara K., Murakami S., Oikava S., Wakamatsu S. Wind tunnel experiments on how thermal stratification affects flow in and above urban street canyon // Atmospheric Environment. 2000. V. 34. P. 1533.

[4]. Kastner-Klein P., Fedorovich E., Rotach M.W. A wind tunnel study of organised and turbulent air motions in urban street canyons // Journal of Wind Engineering and Industrial Aerodynamics. 2001. V. 89. P. 849–861.

Investigation of the effect of the length of the street on the flow of air in them

[1,2] *M. Volik <volikmv@mail.ru>*
[1] *Financial University under the Government of the Russian Federation, 7, Molodegnaya st. , Republic of North Ossetia-Alania, Vladikavkaz, 362002, Russia*
[2] *South Mathematical Institute of the Vladikavkaz Scientific Center of the Russian Academy of Sciences and the Government of the Republic of North Ossetia-Alania, 22, Marcus st., Republic of North Ossetia-Alania, Vladikavkaz, 362027, Russia*

Abstract. The paper compares the results of mathematical modeling of aerodynamics typical urban developments with different length of the streets. The calculations were performed using the redistributable package OpenFoam and remote console access to the control node computing cluster BL2x220 Cluster Console https://unihub.ru/resources/bl2x220cc Web-laboratory Unihub (www.unihub.ru) program "University cluster" (www.unicluster.ru). Consider a single street with houses of the same height at its sides, a single street with houses of different heights at its sides and two parallel streets located on a hillside. Comparison of simulation results in two-dimensional and three-dimensional approximation showed that the length of the street has a significant impact on the qualitative and quantitative flow pattern inside the streets and on buildings.

Keywords: mathematical modeling; aerodynamics; urban canopy; OpenFoam.

References

[1]. Kraposhin M.V., Samovarov O.I., Strizhak S.V. Paket OpenFoam: chislennoe modelirovanie zadach MSS [Package OpenFoam: numerical simulation of continuum mechanics] // Materialy shkoly-seminara «Osnovy ispol'zovaniya OpenFoam, Salome, ParaView» [Materials the school-seminar "Basics of using OpenFoam, Salome, ParaView»], https://unihub.ru/tools/unicfdc1/svn/trunk/Version2/Pdf/day1_2_4-OpenFOAM-Base.pdf.

[2]. Kamenetskiy E.S., Volik M.V., Tagirov A.M. Matematicheskoe modelirovanie rasprostraneniya zagryaznyayushchikh veshchestv, vybrasyvaemykh avtotransportom [Mathematical modeling of pollutants dispersion emitted by cars] // Izvestiya Kabardino-Balkarskogo nauchnogo tsentra RAN [News Kabardino-Balkar Scientific Center of RAS], № 6(62), 2014 g. – s.23-32.

[3]. Uehara K., Murakami S., Oikava S., Wakamatsu S. Wind tunnel experiments on how thermal stratification affects flow in and above urban street canyon // Atmospheric Environment. 2000. V. 34. P. 153.

[4]. Kastner-Klein P., Fedorovich E., Rotach M.W. A wind tunnel study of organised and turbulent air motions in urban street canyons // Journal of Wind Engineering and Industrial Aerodynamics. 2001. V. 89. P. 849–861.