

## 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, категоризируемое 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 итерированного 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 решения 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  $\text{while}(\dots)$ , например, итерационные методы решения СНУ). Вершины орграфа отождествляются при этом с серверами облака, раздающими исходные данные и иницилирующими 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  $G^{(0)} = (V^{(0)}, D^{(0)}, \Gamma^{(0)})$ , где  $V^{(0)}$  – набор вершин,  $D^{(0)}$  – набор дуг,  $\Gamma^{(0)}$  – весовые характеристики дуг. При этом  $n^{(0)} = |V^{(0)}|$  – число вершин,  $m^{(0)} = |D^{(0)}| = |\Gamma^{(0)}|$  – 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:

$$(G^* \equiv (V^*, D^*, \Gamma^*)) = \underset{G}{\text{Arg min}} \theta(G(G^{(0)})), \quad (1)$$

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

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

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

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

$$\gamma_k = I(d_k) \times \frac{\bar{E}_{\theta,k} + \theta(d_k)}{\theta(d_k)} \times \frac{\bar{E}_{T,k} + [T(d_k)]^{1_k}}{T(d_k)}, \quad (2)$$

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 полнота 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 результирующей параметричности минимальна.

2.  $\bar{E}_{T,k} \geq 1$  – параметр, определяющий рост параметричности  $\gamma_k$  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  $\text{Str}(S)$  – является отображением, ставящим в соответствие системе  $S$  взвешенный орграф  $G$

$$\text{Str}(S): S \longrightarrow G. \quad (3)$$

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

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

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

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

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

где « $\text{des}(\circ)$ » – характеристикация, 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;
- $D = (d_k = \text{des}(v_i \rightarrow v_j), k = 1..m; i, j \in \{1, n\}, i \neq j)$  – кортеж дуг;

–  $\Gamma = (\gamma_k, k = 1..m; \gamma_k := d_k)$  – кортеж параметричностей; функтор  $\gamma(d_k)$  переопределяет  $\gamma_k$  и является процедурной моделью процесса параметризации дуг;  $\gamma_k := d_k$  означает « $\gamma_k$  соответствует  $d_k$ »<sup>1</sup>.

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  $d_k = des(v_i \rightarrow v_j)$  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

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

## 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.

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

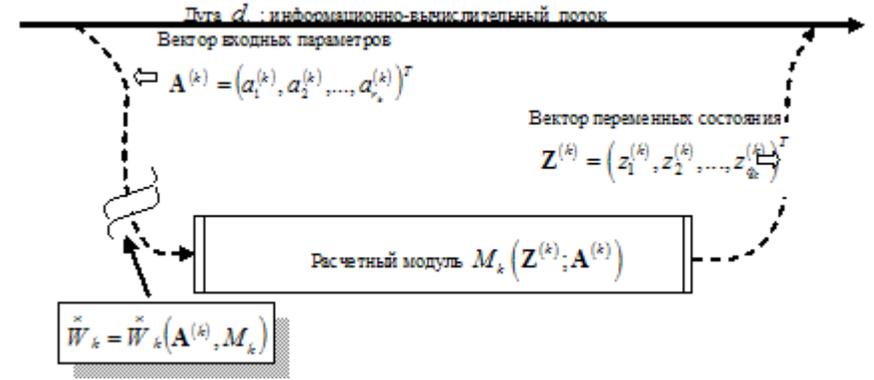


Рисунок 1. Вычислительная нагрузка дуги орграфа

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  $M_k(\mathbf{Z}^{(k)}; \mathbf{A}^{(k)})$ , 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) результирующая информационно-вычислительная нагрузка дуги в условиях разрыва  $\overset{\times}{d}_k := \overset{\times}{\gamma}_k$  может отличаться от обычного состояния  $\bar{d}_k := \bar{\gamma}_k$ ; 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  $\gamma_k = \circ \left( \bar{\gamma}_k, \gamma_k^* \right)$ , где

$\circ$  – функциональный абстрактор<sup>2</sup>.

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

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<sup>3</sup> в виде предиката  $\bar{W}_k = \bar{W}_k(\mathbf{A}^{(k)}, M_k) \in \{0,1\}$ . Значение «0» соответствует продолжению итераций, а «1» – окончанию цикла. В случае, когда все  $\bar{W}_k = 1, k = 1..n^*$ , считаем, что вычислительная нагрузка в системе равна нулю (здесь  $n^*$  – 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  $\Gamma$ . 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<sup>4</sup>

<sup>2</sup> 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».

<sup>3</sup> В частности, такой моделью может быть тот или иной итерационный метод решения систем уравнений.

<sup>4</sup> В формуле (7) и далее по тексту используется обозначение  $\complement$  – «в противном случае».

$$\left( \left\{ \begin{array}{l} G = \text{con}(\mathbf{X}), \\ \underline{G} = \text{con}(\underline{\mathbf{X}}) \end{array} \right\}, \left\{ \begin{array}{l} x_{ij} = \text{des}(\gamma_k := (i \rightarrow j)), \\ x_{ij} = \text{des}(\exists(i \rightarrow j) \Rightarrow 1 \complement 0), \\ x_{ij} = \text{sign}(x_{ij}) \end{array} \right\}; i, j = 1..n \right) = (1 := \text{true}). \quad (7)$$

*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  $\alpha$ , называются сопоставимыми и это обозначается  $G_1 \cong G_2$ , если выполняется условие

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

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

$$\theta(G_2) = \nu(\alpha) \theta(G_1), \nu(\alpha) : \forall \alpha > 0 : \nu(\alpha) < \nu(\bar{\alpha}), \bar{\alpha} > \alpha. \quad (10)$$

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

накладывается ограничение в виде двойного неравенства<sup>5</sup>. 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 связанных 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 связанных 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  $R(G): \exists G', G' \neq G \Rightarrow R(G) = R(G') \wedge \forall G' \subset G: R(G) > R(G')$ . 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

<sup>5</sup> 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  $m$ .

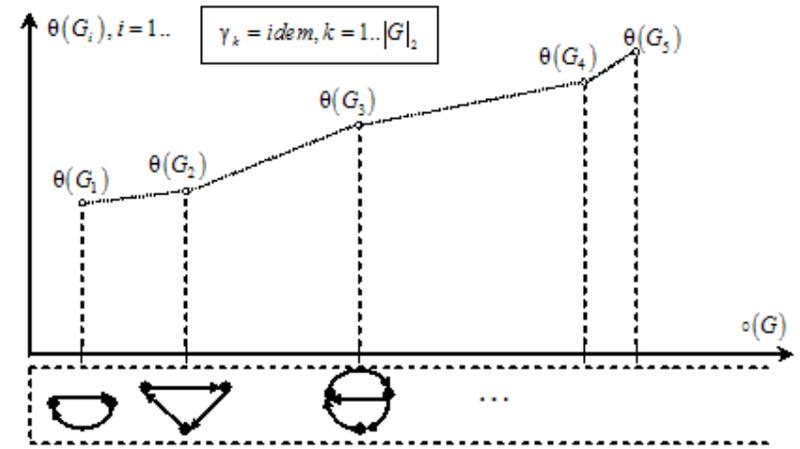


Рисунок 2. Иллюстрация к гипотезе о росте оценок сложности орграфов при их лексиграфическом перечислении в условиях постоянства параметрических

$$R(G) = R_0(G) + R_1(G), R_0(G) = 0 + \left[ \begin{matrix} n & n \\ i=1 & j=1 \end{matrix} \left( \left\{ \begin{matrix} i=j \Rightarrow \dots \\ i \neq j \Rightarrow \dots x_{ij} \end{matrix} \right\} \right) \right], R_1(G) = \sum_{i=1}^{|G|_2} 2^{i+k-1}, \quad (11)$$

$$k = \left\lfloor \log_2 R_0(\hat{G}) + \frac{1}{2} \right\rfloor, \hat{G} = \text{con}(\hat{\mathbf{X}}), G = \text{con}(\mathbf{X}),$$

$$\dim \mathbf{X} = \dim \hat{\mathbf{X}}, \hat{x}_{ij} = 1, i = 1..n, j = 1..n, i \neq j, n = |G|_1$$

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

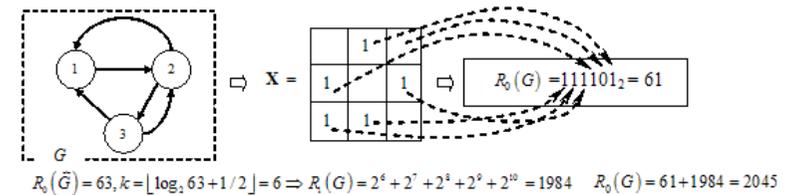


Рисунок 3. Пример вычисления инварианта орграфа

#### 4. Структурная декомпозиция орграфа ОВС

В общем случае изначально орграф ОВС может содержать сильно связанные 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 связанный; 2) орграф  $G$  не является сильно связанным. В первом случае матрица достижимости

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

заполнена единицами полностью, во втором – лишь частично.

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

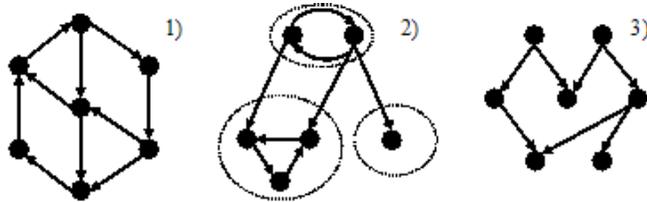


Рисунок 4. Три варианта орграфа (бикомпоненты – в овалах)

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 связанного 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<sup>6</sup>, если в нем отсутствуют контуры и, чтобы

<sup>6</sup> 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

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

*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

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

*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

$$\hat{G}' \subseteq \hat{G} \Rightarrow \theta(\hat{G}') \leq \theta(\hat{G}). \quad (15)$$

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

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

Вариант № 2 – оценка сложности дерева с учетом загруженности путей

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

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  $P = \{P_k, k = \overline{1, p}\}, P_k = P_{k,1} \rightarrow P_{k,2} \rightarrow \dots \rightarrow P_{k,|P_k|}; P_{k,j}$  – дуга, а  $\tilde{\gamma}(P_{k,j})$  – ее вес.

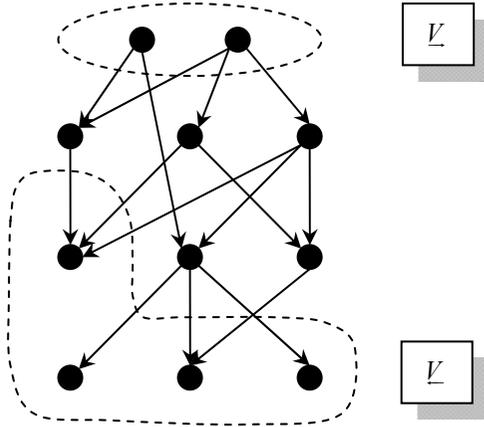


Рисунок 5. Иллюстрация к формуле (17)

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  $\theta^{(1)}(\hat{G})$  – 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  $\theta^{(1)}(\hat{G})$  на множестве вариантов схем распараллеливания.

2) оценка  $\theta^{(2)}(\hat{G})$  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 загрузке 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 связанных подграфов (ССП).

Для оценки сложности деревьев ССП (сокращенно ДССП) необходимо преобразовать ДССП в дерево обобщенных вершин – оргграф 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<sup>7</sup> (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 оргграф 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  $\tilde{X} : \tilde{X} = \text{con}(G), \tilde{X} := \underline{G}, G \approx \underline{G}, \tilde{X} := \mathbf{X}$ , где  $\mathbf{X}$  – традиционная, не взвешенная

<sup>7</sup> Альтернативой является нагрузка на вершины в виде потенциала вершин. На наш взгляд, такой подход может сильно усложнить оценку сложности, особенно, когда оргграф 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  $\ddot{X}$  матрицы смежности  $X$  верифицируется подстановкой в следующее операторное уравнение (если она записана в терминах языка спецификаций, разумеется)

$$\textcircled{1}: \lambda(\beta) \equiv \left\{ bic(\tilde{G}_i \setminus \tilde{D}_i) = 1, i = 1.. \beta \right\} : \left( \bigcup_{i=1}^{\beta} (\tilde{G}_i \cup \tilde{D}_i) = G \right) = 1 \wedge$$

$$\forall \beta_1 < \beta : \lambda(\beta_1) = 0, \tilde{D}_i := \tilde{G}_i, \tilde{D}_i = (d_j^{(i)}, j = 1.. \gamma_i), \ddot{X}^T := \bigcup_{i=1}^{\beta} (\tilde{G}_i \cup \tilde{D}_i), \quad (18)$$

$$\textcircled{2}: \tilde{G}_1 < \tilde{G}_2 < \dots < \tilde{G}_\beta : \forall j \in \{1, \beta-1\} : \exists l > j : \exists k, \left[ (d_k = (v_{k_l} \rightarrow \circ)) : v_{k_l} \in \tilde{G}_l \right],$$

где  $\lambda(\beta)$  –  $\lambda$ -предикат, по которому осуществляется проверка операторного уравнения на тождество. Использование  $\lambda$ -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); « $\setminus$ » – операция вычитания множеств;  $\gamma_i$  – дуговая размерность подмножеств дуг, образующих подматрицы в квазитреугольной форме  $\ddot{X}$ ;  $\tilde{G}_1 < \tilde{G}_2 < \dots < \tilde{G}_\beta$  – условие лексиграфического построения  $\ddot{X}$ , а именно – над-квазидиагональ должна быть нулевой.

Если процедурная модель построения матрицы  $\ddot{X}$  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<sup>8</sup> 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  $\textcircled{1}$  и  $\textcircled{2}$ ;

<sup>8</sup> 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<sup>9</sup>.

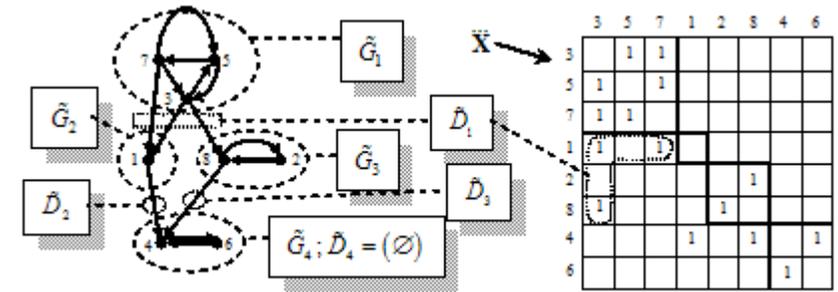


Рисунок 6. Иллюстрация к операторному уравнению (18)

Рассмотрим подробности (18) – сначала уравнение  $\textcircled{1}$ , состоящее из двух частей, объединенных логическим «И». В первом принимает участие  $\lambda$ -предикат и проверяется равенство  $\lambda(\beta) = 1$ . Аргумент  $\beta$  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  $\textcircled{1}$ . 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 максимальной  $\beta$ ; 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)  $\ddot{X}^T := \bigcup_{i=1}^{\beta} (\tilde{G}_i \cup \tilde{D}_i)$  – 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

<sup>9</sup> Если считать допустимым термин «экспериментальная верификация». Фактически это экспериментальная апробация языка спецификаций на стадии его разработки.



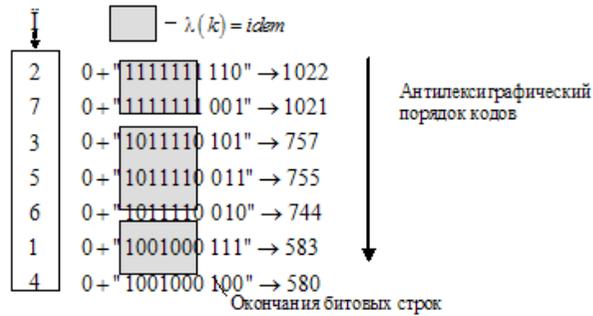


Рисунок 8. Иллюстрация к процедурной модели (19)

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, заключается 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  $\hat{G}_1$  обнаружены две контурные подсистемы  $\hat{G}_1$  и  $\hat{G}_2$ . 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<sub>1</sub> и Ц<sub>2</sub>, а точка сочленения ТС при этом не теряет своей значимости как координатор вычислений. В

данном случае оценка сложности  $\theta(\hat{G}_1 \cup \hat{G}_2) = \theta(\hat{G}_1) + \theta(\hat{G}_2)$  – 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 результирующем 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.

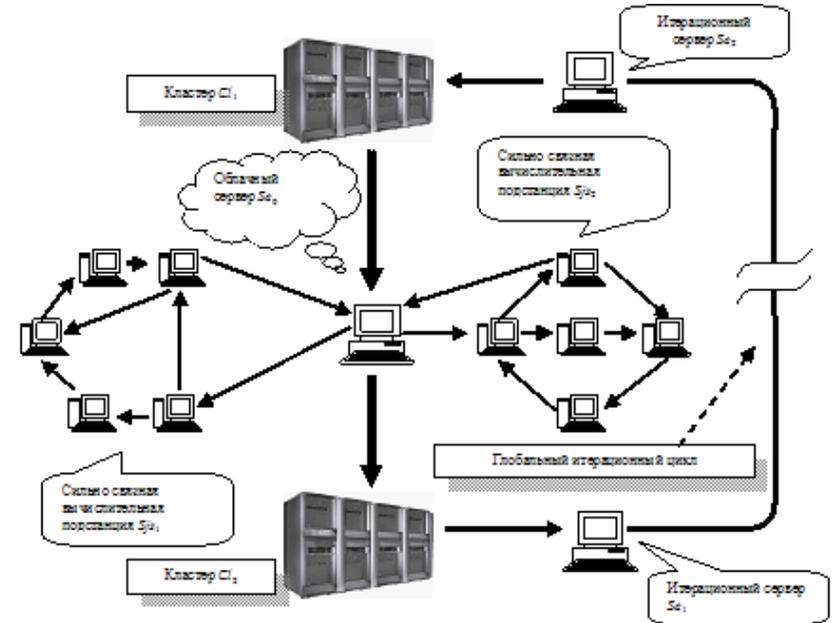


Рисунок 9. Пример ОВС с глобальными итерациями.

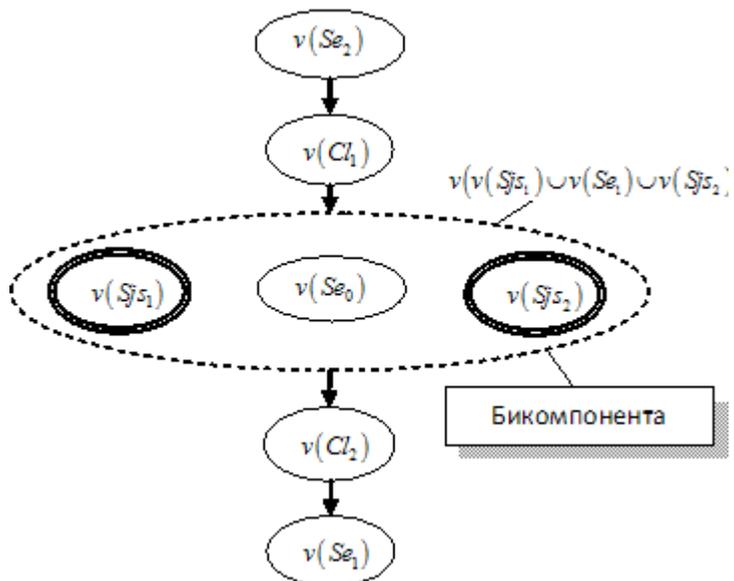


Рисунок 10. Структурирование грид-системы: случай одной бикомпоненты.

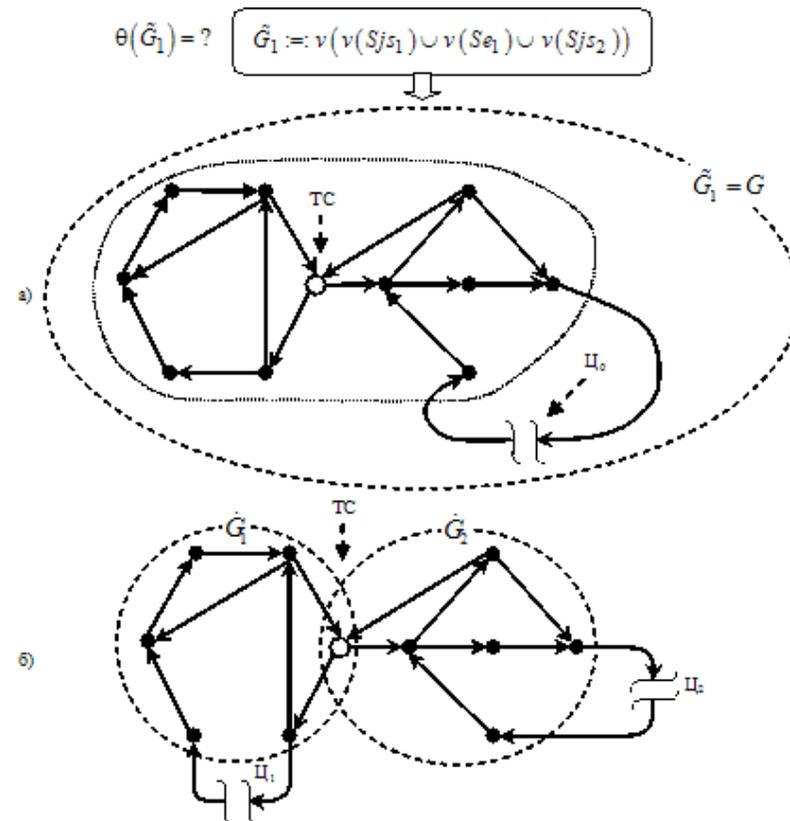


Рисунок 11. Варианты структуризации ОВС на рисунке 10: а) точка сочленения не учитывается; б) учитывается.

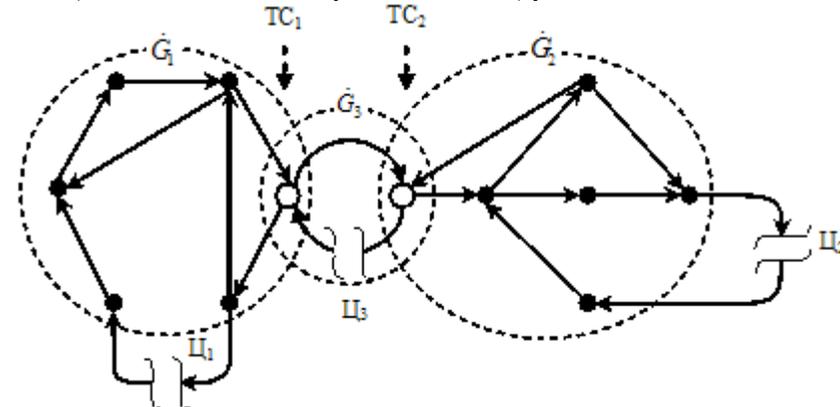


Рисунок 12. Вариант лепестковой структуризации с центральным облачным сервером.

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.

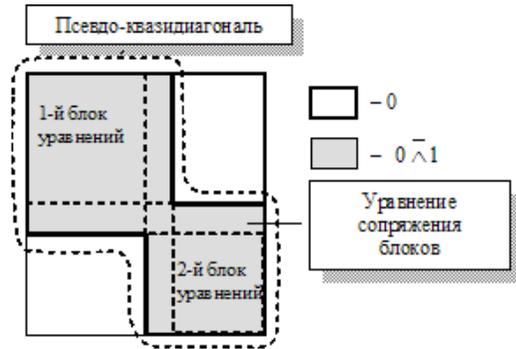


Рисунок 13. Матрица смежности для системы уравнений большой размерности с уравнением сопряжения блоков

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

$C(G) = (C_i(G) := (d_{1..n_i}^{(i)} | \lambda(d_{1..n_i}^{(i)}) = 1), i = \overline{1, K})$  – кортеж контуров орграфа  $G$ ,

где  $K$  – число контуров; в  $i$ -том контуре содержится  $n_i$  дуг,  $\lambda(d_{1..n_i}^{(i)})$  –  $\lambda$ -предикат, в который вложено определение «элементарный контур», а именно

$$\lambda(d_{1..n_i}^{(i)}) = \left( \begin{array}{l} (\overline{d}_k^{(i)} = \underline{d}_{k+1}^{(i)}, k = \overline{1, n_i - 1} \wedge \underline{d}_1^{(i)} = \overline{d}_{n_i}^{(i)}) \wedge \\ \forall v_o \in \overline{d}_k^{(i)}, k \in \{\overline{1, n_i - 1}\} : \exists j \neq k + 1 : v_o = \underline{d}_j^{(i)} \wedge \\ \underline{d}_1^{(i)} < \underline{d}_{2..n_i}^{(i)} \end{array} \right) \Rightarrow 1 \in \mathcal{C}_0, \quad (20)$$

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

$$\begin{aligned} \textcircled{1}: \lambda(\beta) &\equiv \left\{ \left( bic(\dot{G}_i) = 1, i = \overline{1, \beta} \right) : \left( \bigcup_{i=1}^{\beta} \dot{G}_i = G \right) \right\} = 1 \wedge \forall \beta_1 < \beta : \lambda(\beta_1) = 0, \\ \textcircled{2}: \forall j, l \in \{\overline{1, \beta}\}, l \neq j : C_{1..n_j}(\dot{G}_j) \cap C_{1..n_l}(\dot{G}_l) &= \emptyset, \\ \textcircled{3}: \dot{G}_i < \dot{G}_{i_2..i_\beta} : \forall j \in \{\overline{1, \beta-1}\}, \forall l \in \{\overline{2, \beta}\}, l > j : C_i(\overline{\dot{G}_l}) &> C_i(\overline{\dot{G}_j}). \end{aligned} \quad (21)$$

В уравнении « $\textcircled{1}$ » содержатся: требование сильной связности подграфов, условие полноты (объединение подграфов равно исходному орграфу  $G$ ) и рекурсивное условие  $\forall \beta_1 < \beta : \lambda(\beta_1) = 0$ , по которому число искомым

подсистем было максимальным. В уравнении « $\textcircled{2}$ » 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 « $\textcircled{3}$ » 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  $C_i(\overline{\dot{G}_j})$  – 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  $G'$ .

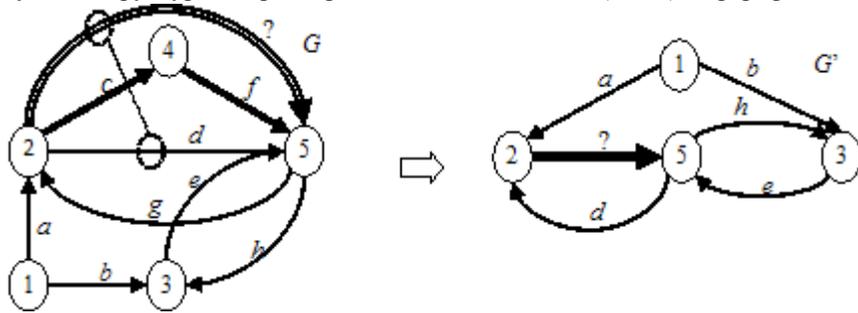


Рисунок 14. Пример орграфа ОВС с избыточными элементами

Транзитивный вычислительный поток  $2 \rightarrow 4 \rightarrow 5$  (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  $2 \rightarrow 5$  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

$$d^* := \left( \underline{d}^* \rightarrow \bar{d}^* \right) = \text{con} \left( \begin{array}{l} \xrightarrow{i=1}^{m^*} (d_i \in D^* : E(D^*) = 1), \\ \gamma^* = \min_{1 \leq i \leq m^*} \gamma(d_i), \underline{d}^* = d_1 \in D^*, \bar{d}^* = d_{m^*} \in D^* \end{array} \right), \quad (22)$$

где утверждается, что обобщенная дуга представлена начальной вершиной  $\underline{d}^*$  и конечной вершиной  $\bar{d}^*$ , 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  $\check{D}^*$ , без которых кортеж  $D^*$  становится транзитивной ветвью

$$E(D^*) = \left( \exists \check{D}^* \in D^* : E(D^* \setminus \check{D}^*) = 1 \wedge |\check{D}^*| = \min |\check{D}^*| \right) \Rightarrow \check{D}^* \subset \emptyset. \quad (2)$$

(3)

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

$$Q(d^*, d_k) = \left( \underline{d}_1^* = \underline{d}_k \wedge \bar{d}_m^* = \bar{d}_k \right) \Rightarrow 1 \subset 0. \quad (24)$$

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

$$d^* \otimes d_k : (d^*, d_k) \rightarrow \left( \underline{d}_k \rightarrow \bar{d}_k \right) = \text{con} \left( (d^*, d_k) \wedge (\gamma^* := \gamma^* + (\gamma_k = \text{des}(d_k) \in G')) \right). \quad (2)$$

(5)

Следует отметить, что 1) перед СПМ матрица контуров  $\check{C}(G) = (\check{c}_{ij}(G))_{K \times m}$  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  $E(D^*)$ : его работа иллюстрируется на рисунке 15.

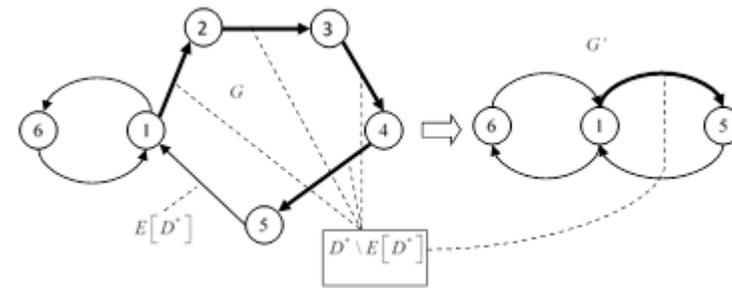


Рисунок 15. Иллюстрация к функтору  $E(D^*)$ .

Сформулируем процедурную модель метода СПМ как рекурсивный алгоритм.

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

- 2°. Инициализируем возвращаемое значение StrMin – оргграф  $G' := G$ .
- 3°. Обнуляем кортеж дуг, составляющих транзитивную ветвь:  $D^* := \emptyset$ ,  $m^* := \emptyset$ .
- 4°. Инициализируем булев массив  $pD^* = [pD_j^* := 0, j = 1..m]$ . Если  $pD_j^* = 1$ , столбец с номером  $j$  считается «рассмотренным» и пропускается.
- 5°. Цикл по не «рассмотренным» столбцам матрицы  $\check{C}(G)$ ,  $j = (1..m) \wedge pD_j^* = 0$ .
- 6°. Цикл по остальным столбцам матрицы  $\check{C}(G)$ , кроме столбца с номером  $j: v = 1..j-1..j+1..m \wedge pD_v^* = 0$ .
- 7°. Проверяем идентичность столбцов  $\check{c}_{ij} = \check{c}_{iv}, i = 1..K$ ? Если они идентичны, добавляем дугу в кортеж  $D^* := D^* \cup \left\{ d_{++m}^* := d_{D^* = \emptyset \Rightarrow j \in v} \right\}$  и помечаем соответствующий столбец как «рассмотренный»:  $pD_{D^* = \emptyset \Rightarrow j \in v}^* = 1$ .
- 8°. Конец цикла по  $v$ .
- 9°. Конец цикла по  $j$ .
- 10°. Тестируем найденный кортеж на транзитивность  $E(D^*) \neq \emptyset$ ? Если тестирование прошло удачно, переходим на п. 12°.
- 11°. Применяем функтор  $E(D^*)$ : если  $E(D^*) \neq \emptyset$ , полагаем  $D^* := D^* \setminus E(D^*)$ , в противном случае процедурная модель заканчивает свою работу с кодом удачи.
- 12°. Исключаем транзитивную ветвь из результатного орграфа  $G' := G' \setminus \lambda G(\circ, D^*, \circ)$  (здесь  $\lambda G(\circ, D^*, \circ)$  – временный оргграф, в котором множество дуг представлено множеством  $D^*$ ) и находим обобщенную дугу

- $d^*$  из решения операторного уравнения (22), добавляем информацию в результатный оргграф  $G' := G' \cup \lambda G(\circ, \{d^*\}, \circ)$ .
- 13°. Проверяем наличие скрытой параллельной ветви, открываем цикл по перебору дуг орграфа  $G'$ , всех, кроме  $d^*$ ,  $k = 1..|G'|_2, d_k \neq d^*$ .
- 14°. Дуга  $d_k$  содержит параллельные вычисления с обобщенной дугой  $d^*$ ? т.е.  $Q(d^*, d_k) = 1$ ? 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:  $G' := \text{StrMin}(G')$ .
- 17°. Выход – оргграф  $G'$ .

## 8. Оценка сложности сильно связанных 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 связанный вычислительный примитив» будет заменен «простейшим сильно связанным оргграфом» 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  $\theta(G)$  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 связанные оргграфы небольшой размерности – из 2-х и 3-х вершин. Для них потребуются вывести формулы оценки сложности.

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

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

Рассмотрим коды всех возможных сильно связанных 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 связанных 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.

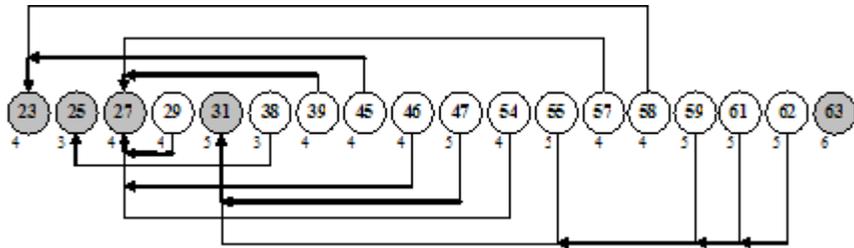


Рисунок 16. Коды  $R_0(G)$  орграфов с тремя вершинами и число дуг (внизу)

## 9. Сложность диполя

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

$$G^{(3)} = (V, D, \Gamma), V = (v_1, v_2), D = (d_1, d_2), \quad (26)$$

$$d_1 = (v_1 \rightarrow v_2), d_2 = (v_2 \rightarrow v_1), \Gamma = (\gamma_1 := d_1, \gamma_2 := d_2).$$

Диполь  $G^{(3)}$  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).

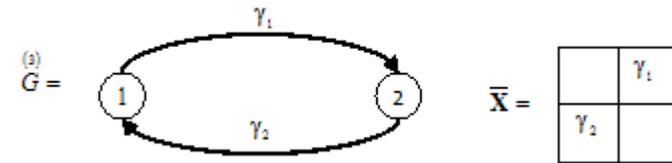


Рисунок 17. Диполь и его взвешенная матрица смежности

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

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

Структура исходного состояния ОВС представлена сильно связанным орграфом

$G^{(3)} \neq G, bic(G) = 1$ . 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 соотношением 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

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

## 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 «контурин» – 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<sup>10</sup> активирована? Нет – посылка сообщения на активизацию.
3. Орграф сильно связный?  $bic(G) = 1$ ? «Да» – на п. 12.
4. Структурная декомпозиция: разбиваем  $G$  на ССП,  $i = 1.. \beta$ .
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:  $\theta$ .

<sup>10</sup> 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 индексируется 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  $\theta$ .
13. Проводим СПМ орграфа  $G = \text{StrMin}(G)$ .
14. Построение контуриона на основе  $\check{C}(G)$  и  $\Gamma$ .
15.  $\theta_0 = \text{HUGE\_VAL}$  (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):  $G = \text{орграф } G'$
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 regionalnykh obrazovatelnykh kompyuternykh 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 regionalnykh kompyuternykh setey / V.E. Podolskiy, Tolstyh S.S. [The use of criteria of the structural complexity for simulation of regional computer networks] // Parallelnyye vyichisleniya v zadachah matematicheskoy fiziki: Sb. st. – Rostov n/D: Izd-vo RGU, 2005. – S. 67 – 75. (In Russian)
- [3]. Podolskiy, V.E. Optimizatsiya klasternykh vyichisleniy s ispolzovaniem kriteriev strukturnoy slozhnosti [Optimization of cluster computing using the criteria of structural complexity] / V.E. Podolskiy, S.S. Tolstykh // Vtoraya Sibirskaya shkola-seminar po parallelnym vyichisleniyam. – Tomsk: Izd-vo Tom. un-ta, 2004. – S. 45-50. (In Russian)