

В.А. Мелентьев, канд. техн. наук (melva@isp.nsc.ru)
(Институт физики полупроводников им. А.В. Ржанова СО РАН,
г. Новосибирск)

ФОРМАЛЬНЫЙ МЕТОД СИНТЕЗА ОПТИМАЛЬНЫХ ТОПОЛОГИЙ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ, ОСНОВАННЫЙ НА ПРОЕКТИВНОМ ОПИСАНИИ ГРАФОВ

Предлагается детерминированный метод синтеза топологий интерконнекта, оптимизированных под требуемые свойства. Метод основан на оригинальном описании графов проекциями, на установлении биективного соответствия требуемых свойств и свойств проекций начального графа, на постулировании соответствующих ограничений модифицируемых проекций и на итеративном применении к ним этих ограничений либо до момента решения системы проекций и получения проекций искомого графа, либо до выявления ее несовместности с заданными начальными условиями.

Ключевые слова: топология интерконнекта; метод проективного описания графов, проекции графа, решение системы проекций и синтез оптимальных топологий.

Введение и постановка задачи

Эффективность параллельной реализации задачи на вычислительных системах (ВС) определяется их топологической совместимостью, оцениваемой степенью соответствия топологии задачи топологиям интерконнекта, т.е. допускаемой последними долей используемых задач вычислительных ресурсов рассматриваемой системы [1]. Понятно при этом, что 100-процентной совместимостью для любых задач обладала бы ВС с полносвязным интерконнектом (если бы таковой мог быть реализован), и в этом случае можно было бы говорить о ее полной универсальности. Максимального использования вычислительных мощностей ВС для критически важных задач можно добиться, реализуя адекватную этим задачам топологию интерконнекта. Однако, и быстродействие и загруженность такого суперкомпьютера существенно снизятся при попытке его использования в решении задач, топологически отличных от его специализации. Поэтому в данной работе речь идет о системах универсального назначения, интерконнект которых, соответственно, также должен обладать некоторыми универсальными топологическими свойствами: регулярностью и компактностью [2-4].

Общепринятой базовой детерминированной моделью топологии ВС является граф, множества вершин и ребер которого взаимно-однозначны, соответственно, множествам вычислительных модулей и линий связи

между ними, а метрические характеристики (диаметр, эксцентриситет, расстояние между вершинами) пропорциональны задержкам передачи/приема единицы информации [5]. Поэтому синтез топологии ВС, оптимальной в отношении сетевых задержек, сводится к синтезу графа с метриками, определяемыми масштабностью решаемых на системе параллельных задач (вычислительными объемами и объемами обрабатываемых данных) и их информационными топологиями. В работах [6, 7] предложена и достаточно подробно рассматривается модель параллельных вычислений, устанавливающая формальные соответствия характеристик параллельных задач метрикам графа ВС.

В связи с этим возникает проблема синтеза графа с требуемыми для оптимальной реализации параллельных алгоритмов метрическими характеристиками. Проблема эта является *NP*-полной и решается, как правило, эмпирически либо с использованием стохастических, комбинаторных, генетических или других эвристических по сути алгоритмов, однако, во-первых, такие решения все равно достаточно трудоемки и, соответственно, времязатратны, а во-вторых, не гарантируют безусловно оптимального в отношении требуемых метрик результата. И то и другое обусловлено тем, что достоверность вероятностно-оптимального синтеза, определяется либо размером выборки, либо мощностью случайно сгенерированной первоначальной популяции, и связанная с этим недетерминированность может привести к непредсказуемым, в том числе деструктивным, последствиям в процессе функционирования системы [2].

Понятно, что возможность получения детерминированного реальновременного результата может быть связана с выбором математического описания объекта исследования (в нашем случае — графа ВС). Приведем в этой связи гиперболический бытовой пример: для забивания гвоздя необходим инструмент. Использование в качестве инструмента ювелирного молоточка потребует нанесения по гвоздю не одной сотни, а то и тысячи ударов, — соответственным будет и затраченное на это время. Однако использование молотка, соответствующего размеру гвоздя, делает поставленную задачу легко и быстро разрешимой. Приведем более релевантный к рассматриваемой здесь тематике пример: описание логической задачи можно задать совокупностью отдельных высказываний и решить задачу последовательным их перебором. Описание той же задачи таблицей истинности или выражениями логической алгебры и использование

соответствующего формального аппарата даст точное и нетрудоемкое решение за несравнимо меньшее время.

Алгоритмы анализа и синтеза графов традиционно базируются на использовании матричных или списковых описаний, задающих *бинарные отношения* вершин и ребер или дуг. Работа с такими низкоуровневыми отношениями подобна приведенным выше примерам использования неадекватных задачам инструментов, — отсюда и неэкспоненциальное (относительно порядка и размера графа) возрастание трудоемкости и временных затрат. В работах [8-11] был предложен и получил развитие новый метод описания, оперирующий более крупными категориями — проекциями графа. Идея об использовании таких описаний для аналитического синтеза графов с требуемыми метриками впервые была высказана и проиллюстрирована примерами в [2], [12–14]. Вследствие новизны этот метод описания графов пока не получил широкого распространения, однако его апробация в создании информационных систем фармацевтической промышленности [15], в исследованиях сетевых информационных систем [16] и в других, например, в [17-20] исследованиях подтвердила его эффективность. С библиографией исследований, основанных на этом методе описания графов, можно ознакомиться в обзорах [21, 22].

В данной работе идея детерминированного синтеза топологий с заданными свойствами [23] нашла свое формальное воплощение в синтезе описанных в [2] компактных графов ВС, обладающих минимальным диаметром, и следовательно, наименьшей топологической задержкой информационного обмена между ветвями параллельной задачи.

1. Предварительные замечания и определения

В данном разделе представлены необходимые пояснения к используемой в работе оригинальной терминологии и соответствующие определения. Добавим, что речь здесь пойдет о простых (без петель и кратных ребер) неориентированных графах.

Проекция $P(v_j)$ графа $G(V, E)$ представляет собой многоуровневую конструкцию, на нулевом уровне которой расположена вершина $v_j \in V$ — основание, или ракурс проекции; порожденное основанием подмножество вершин первого уровня $V_{1j} \subset V$ содержит все вершины его окружения $\mathcal{N}(v_j) \setminus v_j$ без порождающей его вершины v_j , а i -й уровень ($i \geq 1$) представляет собой совокупность подмножеств вершин, каждое из которых порождено вершиной $(i - 1)$ -го уровня и является окружением этой вершины без вершин, непосредственно предшествующих ей в

данной проекции [2]. Фрагменты проекции, порождаемые вершинами 1-го уровня, называем ветвями проекции.

Отношения непосредственного предшествования и порождения вершин в проекции $P(v)$ есть отношения смежности этих вершин. Число порожденных на i -м уровне подмножеств соответствует числу порождающего их вершин $(i - 1)$ -го уровня. Упорядочив вершины по их непосредственным предшествованиям и порождениям от ракурсной v до j -й вершины i -го уровня v_{ij} , получим маршрут $M(v - v_{ij}) = (v, v_{1x}, \dots, v_{ij})$ из вершины v в вершину v_{ij} , при этом в невзвешенном графе номер i уровня равен длине пути из ракурсной вершины v в любую вершину подмножества V_i находящихся на уровне i вершин. Последовательность вершин, непосредственно предшествующих открывающимся скобкам от v_{ij} до ракурсной вершины v , дает обратный маршрут $M(v_{ij} - v) = M^{-1}(v - v_{ij})$.

Если в проекции графа одна и та же вершина встречается m раз, то ее экземпляр, наиболее приближенный к основанию и расположенный первым слева, считаем оригинальным, а остальные $(m - 1)$ ее копий называем реплицированными, или репликами [14]. Появление реплик в проекции указывает на наличие в графе циклов. Сумма номеров уровней, находящихся в разных ветвях оригинальной вершины и ее реплики, или любых двух реплик одной и той же вершины определяет длину соответствующего цикла.

Проекция $P(v_j)$ графа $G(V, E)$ является полной, если ею определены все вершины и все ребра этого графа [2]. Вершинно полная проекция графа $G(V, E)$ содержит все вершины множества V , но не обязательно является полной, тогда как реберно полная проекция является полной всегда. Минимальное число уровней¹ вершинно полной проекции, порожденных ракурсной вершиной (основанием проекции), определяется эксцентриситетом основания, тогда как минимальное число уровней в полной проекции может быть на единицу большим, если хотя бы пара оригинальных вершин эксцентриситетного уровня смежна. Понятно, что все вершины и все ребра графа будут определены совокупностью даже одноуровневых его проекций, каждая из которых по определению задает все отношения смежности ее основания. С доказательствами изложенного здесь материала можно ознакомиться в работах [8-10].

¹ Далее словосочетание «порожденных основанием проекции» опускаем, подразумевая, что в число уровней входят только уровни, стоящие выше основания (нулевого уровня).

Проиллюстрируем вышесказанное проекциями 3-мерного гиперкуба. Приведение его здесь в виде рисунка считаем избыточным в силу общеизвестности того, что смежность вершин задана их нумерацией: расстояние между вершинами гиперкуба равно двоичному расстоянию между их номерами.

$$P(0) = 0 \left(\begin{matrix} 1^{(3^{(2,7)}, 5^{(4,7)})}, \\ 2^{(3^{(1,7)}, 6^{(4,7)})}, \\ 4^{(3^{(1,7)}, 6^{(2,7)})} \end{matrix} \right) \equiv 0 \left(\begin{matrix} 1^{(3^{(2,7)}, 5^{(4,7)})}, \\ 2^{(3^{(1,7)}, 6^{(4,7)})}, \\ 3^{(5^{(1,7)}, 6^{(2,7)})} \end{matrix} \right),$$

$$P(5) = 5 \left(\begin{matrix} 1^{(0^{(2,4)}, 3^{(2,7)})}, \\ 4^{(0^{(1,2)}, 6^{(2,7)})}, \\ 7^{(3^{(1,2)}, 6^{(2,4)})} \end{matrix} \right) \equiv 5 \left(\begin{matrix} 1^{(0^{(2,4)}, 3^{(2,7)})}, \\ 4^{(0^{(1,2)}, 6^{(2,7)})}, \\ 7^{(3^{(1,2)}, 6^{(2,4)})} \end{matrix} \right).$$

Описание каждой из приведенных выше проекций дано в двух тождественных вариантах: однострочном и многострочном. Во втором варианте число строк соответствует числу ветвей проекции, или степени ракурсной вершины. Такое описание, во-первых, более компактно, а во-вторых, позволяет рассматривать отдельные ветви проекций, что будет использовано нами в дальнейшем. Заметим, что любая 3-уровневая проекция 3-куба является полной, поэтому на данном примере нетрудно убедиться в простоте и нетрудоемкости использования проекций для выявления кратчайших или заданной длины маршрутов, диаметра, эксцентриситета, обхвата и прочих циклов, а также других характеристик описанного таким образом графа.

Компактные графы вычислительных систем.

Одной из насущных проблем управления вычислительными системами (ВС) является критичность к времени реализации управляющих воздействий. Своевременность управляющих воздействий в процессе функционирования ВС во многом определяется компактностью топологии ее интерконнекта, обеспечивающей минимум числа транзитов межпроцессорных взаимодействий. Впервые понятие компактных графов как s -регулярных графов порядка n с минимально возможным диаметром d введено в [2], там же определено условие компактности регулярного графа, связывающее степень s его вершин со значениями n и d :

$$1 + s \sum_{i=1}^{d-1} (s-1)^{i-1} < n \leq 1 + s \sum_{i=1}^d (s-1)^{i-1} \quad (1)$$

Далее удовлетворяющие этому условию графы будем называть $n(s,d)$ -компактными, и задача построения оптимальных в отношении задержек топологий ВС состоит в синтезе соответствующих условию (1) графов.

Синтез графов с требуемыми свойствами

Требуемые свойства искомого графа $G(V,E)$ совершенно однозначно отображаются в свойствах его проекций, например, число вершин $n = |V|$ графа G равно числу его проекций $|P(G)|$, значение диаметра d графа указывает на то, что число уровней любой его вершинно полной проекции не превышает d и т. п. Таким образом, если синтезируемый граф G должен обладать некоторым множеством требуемых свойств $Q(G)$, то это множество должно быть биективно и множеству свойств его проекций $Q(P(G))$: $f: Q(G) \leftrightarrow Q(P(G))$. Очевидно, что и свойства $Q(P(v_i))$ каждой проекции $P(v_i) \in P(G)$ должны быть тождественны свойствам $Q(P(G))$: $Q(P(G)) \rightarrow Q(P(v_i))$. Тогда, построив некий связный исходный граф $G'(V,E')$ порядка n с меньшим чем $|E|$ числом ребер $E' < E$, свойства проекций $Q(P'(v_i))$ которого не противоречат требуемым свойствам $Q(P(G))$, и достраивая его ребрами так, чтобы это условие непротиворечивости сохранилось для каждой проекции при каждом E' вплоть до $|E'| = |E|$, получим искомый граф $G(V,E)$, если таковой существует.

Покажем, например, общие свойства искомого n -вершинного s -регулярного компактного графа с заданным диаметром d и соответствующие им свойства проекций. j

$Q(G)$	$Q(P'(G))$
Граф $G(V,E)$ — связный степени s и диаметра d	Каждая d -уровневая проекция $P(v_i) \in P(G)$ является вершинно полной и содержит все вершины из V
Порядок графа $G(V,E)$ — $n = V$	Число реплик c в d -уровневой проекции — $c = 1 + s \sum_{i=1}^d (s-1)^{i-1} - n$.
Граф $G(V,E)$ не содержит кратных ребер и петель	Первый уровень каждой проекции не содержит реплик

Если существует $n(s,d)$ -компактный граф с числом вершин $n = 1 + s \sum_{i=1}^d (s-1)^{i-1}$, то называем его предельно (s,d) -компактным, и к приведенным выше свойствам $Q(P'(G))$ добавим: любая d -уровневая проекция такого графа не содержит реплик, и $c = 0$.

Рассмотрим простой демонстрационный пример синтеза предельно $(3,2)$ -компактного ($s=3, d=2$) графа. Из условия (1) получим $n = |V| = 10$,

Построим двухуровневую проекцию $P(v_0)$ остовного дерева¹ искомого графа $G(V,E)$ соответствующую диаметру $d = 2$, пронумеровав в ней вершины v графа от $v = 0$ до $|V| - 1 = 9$: корневой вершине этого дерева (основанию его проекции) присвоим нулевое значение, остальные вершины в проекции $P(0)$ для определенности нумеруем по порядку снизу вверх и слева направо. Так как по условию данной задачи искомым граф должен быть предельно компактным, то каждая двухуровневая вершинно полная его проекция содержит только оригинальные вершины и не содержит реплик ($c = 0$).

$$P(0) = 0 \begin{pmatrix} 1^{(4,5)} \\ 2^{(6,7)} \\ 3^{(8,9)} \end{pmatrix} \quad (2)$$

Подмножества вершин, заполняющих l -уровень j -ветви проекции $P(v)$ обозначим $M_{l,j}(v)$, здесь $l \in (1, \dots, d), j \in (1, \dots, s)$; так, подмножества первого уровня — $M_{1,1}(0) = (1)$, $M_{1,2}(0) = (2)$, $M_{1,3}(0) = (3)$ — содержат всего по одному элементу, подмножества второго — $M_{2,1}(0) = (4,5)$, $M_{2,2}(0) = (6,7)$, $M_{2,3}(0) = (8,9)$ — содержат по два элемента. Аналогично при поиске, например, предельно (3,3)-компактного графа, где $n = 22$: третий уровень проекции $P(0)$ составляли бы подмножества $M_{3,1}(0) = (10,11,12,13)$, $M_{3,2}(0) = (14,15,16,17)$, $M_{3,3}(0) = (18,19,20,21)$.

Используя проекцию $P(0)$, построим остальные проекции $P'(v_i)$, $0 < v_i < |V|$ (здесь штрих в $P'(v_i)$ указывает на ее незавершенность: в ней существуют вакансии², обозначенные символом *). Ниже каждой проекции разместим множество V вершин, в котором вершины уже упомянутые в рассматриваемой проекции скроем (позначим серым цветом), их множество обозначим $\bar{M}^*(v_i)$. Тогда черными останутся лишь не упомянутые в проекции $P'(v_i)$ вершины множества $M^*(v_i) = V \setminus \bar{M}^*(v_i)$, они же — соискатели вакансий. Заметим, что в случае построения предельно компактного графа число вакансий равно числу вершин-соискателей: $c(v_i) = |M^*(v_i)|$.

Проекция $P'(v_i)$, за исключением «нулевой» $P'(v_i) \equiv P(0)$, первоначально не являются остовными и становятся таковыми лишь в

¹ В зависимости от заданных свойств искомого графа $G(V,E)$ исходный граф не обязательно должен быть остовным, но должен содержать все вершины искомого графа, т. е. быть связным суграфом.

² Вакансиями называем пустые (незаполненные) места в проекции, соответствующие вершинам, на текущий момент построения пока еще неизвестным.

процессе построения искомого графа: если такой граф будет найден, то множества $M^*(v_i)$ во всех проекциях $P(v_i)$ станут пустыми (штрих в обозначении проекции здесь опущен, так как в таком случае $P'(v_i) \equiv P(v_i)$). Справа от ветвей и уровней каждой проекции поместим подмножества вершин, запрещенных для замещения вакансий этих ветвей и их уровней. Если вакансии взятых для рассмотрения уровня и ветви отсутствуют, помечаем это символом « \equiv ». Вершины, окружения которых полностью определены изначально (как в $P(0)$) или в процессе построения (см., например, $P'(1)$), вакансий смежности не имеют.

Прежде чем перейти непосредственно к построению проекций и перечней запрещенных в них вакансий построим начальную таблицу смежностей и вакансий вершин исходного графа:

0	0	1	2	3	4	5	6	7	8	9
1	0	1	2	3	4	5	6	7	8	9
2	0	1	2	3	4	5	6	7	8	9
3	0	1	2	3	4	5	6	7	8	9
4	0	1	2	3	4	5	6	7	8	9
5	0	1	2	3	4	5	6	7	8	9
6	0	1	2	3	4	5	6	7	8	9
7	0	1	2	3	4	5	6	7	8	9
8	0	1	2	3	4	5	6	7	8	9
9	0	1	2	3	4	5	6	7	8	9

Здесь номер строки соответствует номеру вершины, номера смежных ей вершин (уже известных вершин ее окружения) даны в полужирном начертании, например, для вершины (строки) **0** — это вершины **1**, **2**, и **3**, для вершины **3** — вершины **0**, **8**, **9** и т. д. Отношения смежности (окружения) вершин 0-3 изначально полностью определены проекцией $P(0)$ — число «жирных» вершин равно заданной степени искомого графа, в рассматриваемом здесь случае — трем, поэтому вакансии в этих строках отсутствуют, и остальные вершины в них скрыты (имеют серое начертание). Диагональные номера вершин в таблице также скрыты, так как искомый граф по условию не имеет кратных ребер и петель. Как видим, таблица диагонально симметрична, это также определяется условием задачи — неориентированностью выстраиваемого графа.

Итак, приведем $P(0)$ и полученные из нее остальные проекции $P'(v_i) | 0 < v_i \leq 9$ каркаса искомого предельного (3,2)-компактного графа, с соответствующими каждой ветви проекции и ее уровням

подмножествами $\bar{M}_{l,j}^*(v_i)$ вершин, запрещенных для заполнения вакансий в l -й ветви и на j -м уровне. Изначально эти подмножества равны подмножеству $\bar{M}^*(v_i)$ «серых» вершин, извлекаемому из приведенного ниже каждой проекции множества вершин $V = \bar{M}^*(v_i) \cup M^*(v_i)$. По результатам рассмотрения каждой проекции корректируем таблицу смежности и вакансий. Например, из $P'(1)$ видим, что вершины 4 и 5 не могут быть смежны вершинам с нулевой по пятую, поэтому корректируем четвертую и пятую строки таблицы, скрывая в них, соответственно, пятую и четвертую вакансии, не затрагивая при этом вершину **1**, хотя и попавшую в этот интервал (0-5), но уже занявшую одну из трех (степень s -регулярного графа $s = 3$ по условию задачи) вакансий смежности и помеченную жирным шрифтом. Аналогично поступая с остальными проекциями, модифицируем таблицу, и приведем ее под проекциями.

$$\begin{array}{l}
 0 \begin{pmatrix} 1^{(4,5)} \\ 2^{(6,7)} \\ 3^{(8,9)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 1 \begin{pmatrix} 0^{(2,3)} \\ 4^{(*)} \\ 5^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-5} \\ \equiv_{0-5} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 2 \begin{pmatrix} 0^{(1,3)} \\ 6^{(*)} \\ 7^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-3,6,7} \\ \equiv_{0-3,6,7} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 3 \begin{pmatrix} 0^{(1,2)} \\ 8^{(*)} \\ 9^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-3,8,9} \\ \equiv_{0-3,8,9} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 4 \begin{pmatrix} 1^{(0,5)} \\ *^{(*)} \\ *^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ 0-5^{0,1,4,5} \\ 0-5^{0,1,4,5} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 5 \begin{pmatrix} 1^{(0,4)} \\ *^{(*)} \\ *^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ 0-5^{0,1,4,5} \\ 0-5^{0,1,4,5} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 6 \begin{pmatrix} 2^{(0,7)} \\ *^{(*)} \\ *^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ 0-3,6,7^{0,2,6,7} \\ 0-3,6,7^{0,2,6,7} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 7 \begin{pmatrix} 2^{(0,6)} \\ *^{(*)} \\ *^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ 0-3,6,7^{0,2,6,7} \\ 0-3,6,7^{0,2,6,7} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 8 \begin{pmatrix} 3^{(0,9)} \\ *^{(*)} \\ *^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ 0-3,8,9^{0,3,8,9} \\ 0-3,8,9^{0,3,8,9} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 9 \begin{pmatrix} 3^{(0,8)} \\ *^{(*)} \\ *^{(*)} \end{pmatrix} \begin{pmatrix} \equiv \\ 0-3,8,9^{0,3,8,9} \\ 0-3,8,9^{0,3,8,9} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9;
 \end{array}$$

0	0	1	2	3	4	5	6	7	8	9
1	0	1	2	3	4	5	6	7	8	9
2	0	1	2	3	4	5	6	7	8	9
3	0	1	2	3	4	5	6	7	8	9
4	0	1	2	3	4	5	6	7	8	9
5	0	1	2	3	4	5	6	7	8	9
6	0	1	2	3	4	5	6	7	8	9
7	0	1	2	3	4	5	6	7	8	9
8	0	1	2	3	4	5	6	7	8	9
9	0	1	2	3	4	5	6	7	8	9

Следующий этап построения искомого графа состоит в выборе и заполнении одной из незаполненных в таблице вакансий смежности. Выбираем произвольную вершину (строку таблицы) с вакансиями и анализируем их. Строки 0-3 вакансий не имеют, строки 4-9 имеют по две вакансии с четырьмя претендентами на них. Выбираем для определенности первую из этих строк — четвертую. Итак, на две вакансии в четвертой строке претендуют вершины 6-9, причем соответствующие этим вершинам строки таблицы имеют две отличающиеся конфигурации. *Конфигурацией* строки v_i для графа $G'(V,E')$ будем называть кортеж $C'(v_i) = \langle N'(v_i), N'(v_i) \rangle$ из двух непересекающихся подмножеств, первое из которых — упорядоченное по номеру множество $N'(v_i)$ выделенных полужирным шрифтом вершин, смежных вершине v_i , а второе — также упорядоченное по номеру множество $N'(v_i)$ вершин с обычным шрифтом, претендующих на свободные вакансии смежности вершине v_i , иначе потенциально смежных ей. Сравнивая конфигурации строк с номерами 6-9 вершин-соискателей из $N'(4)$, т. е. строк с шестой по девятую, заметим, что шестая и седьмая строки имеют одинаковые конфигурации — $C'(6) = C'(7) = \langle \langle \mathbf{2} \rangle, (4,5,8,9) \rangle$, но эти конфигурации отличаются от также одинаковых конфигураций восьмой и девятой строк: $C'(8) = C'(9) = \langle \langle \mathbf{3} \rangle, (4,5,6,7) \rangle$. То же самое можно отнести и к вершинам 5-9 и к конфигурациям соответствующих строк таблицы потенциальной смежности.

Сформулируем утверждение для рассматриваемого случая, не обобщая его на произвольные случаи. Доказательство этого утверждения будет получено в процессе построения.

Утверждение 1: Если при построении предельно (3,2)-компактного графа вершина v_i имеет две вакансии, и строка v_i содержит четыре

претендующие на них вершины, то ее вакансии смежности будут заполнены вершинами-претендентами с отличающимися конфигурациями, и выбор варианта заполнения вакансий между вершинами с равными конфигурациями может быть произвольным.

Согласно утверждению 1 обе вакансии смежности вершины 4 могут быть заполнены вершиной из (6,7) и вершиной из (8,9). Свобода выбора варианта заполнения вакансии следует из того, что вершины 6 и 7, как и вершины 8 и 9, — висячие и порождены одной и той же вершиной: в первом случае вершиной 2, во втором — вершиной 3. Следовательно, перенумерация вершин 6 в 7 или 8 в 9 никак не изменит рассматриваемый в данный момент граф $G'(V, E')$. Распространив сказанное на вершины 5-9, получим: вершина 5 смежна вершинам 6 или 7 и 8 или 9, вершина 6 — вершинам 4 или 5 и 8 или 9, ..., вершина 9 будет смежна вершинам 4 или 5 и 6 или 7. Заметим, что все эти высказывания относительно вариантов заполнения вакансий смежности вершин 4-9 согласуются между собой и непротиворечивы.

Выбираем для определенности соединение ребром вершин 4 и 6 и модифицируем проекции $P'(v_i) | 0 < v_i \leq 9$, множества $\bar{M}^*(v_i)$ «серых» и $M^*(v_i)$ «черных» вершин, новую таблицу потенциальной смежности дополненную ребром 4-6 поместим ниже обновляемых проекций и, если этого потребует какая-либо из проекций, будем вносить в нее соответствующие изменения.

$$\begin{array}{l}
 0 \begin{pmatrix} 1^{(4,5)} \\ 2^{(6,7)} \\ 3^{(8,9)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 1 \begin{pmatrix} 0^{(2,3)} \\ 4^{(6,*)} \\ 5^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-6} \\ \equiv_{0-6} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 2 \begin{pmatrix} 0^{(1,3)} \\ 6^{(4,*)} \\ 7^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-4,6,7} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 3 \begin{pmatrix} 0^{(1,2)} \\ 8^{(*,*)} \\ 9^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-3,8,9} \\ \equiv_{0-3,8,9} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 4 \begin{pmatrix} 1^{(0,5)} \\ 6^{(2,*)} \\ *^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7,3,7} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 5 \begin{pmatrix} 1^{(0,4)} \\ *^{(*,*)} \\ *^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-6^{0,1,4,5}} \\ \equiv_{0-6^{0,1,4,5}} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 6 \begin{pmatrix} 2^{(0,7)} \\ 4^{(1,*)} \\ *^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7^{0-2,3,4,6-9}} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9; \\
 7 \begin{pmatrix} 2^{(0,6)} \\ *^{(*,*)} \\ *^{(*,*)} \end{pmatrix} * \begin{pmatrix} \equiv \\ \equiv_{0-4,6,7^{0,2,6,7}} \\ \equiv_{0-4,6,7^{0,2,6,7}} \end{pmatrix}, \\
 0,1,2,3,4,5,6,7,8,9;
 \end{array}$$

$$\begin{array}{ccc}
2 \begin{pmatrix} 0^{(1,3)} \\ 6^{(*,*)} \\ 7^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-3,6,7} \\ \equiv_{0-3,6,7} \end{pmatrix}, & \Longrightarrow & 2 \begin{pmatrix} 0^{(1,3)} \\ 6^{(4,*)} \\ 7^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-4,6,7} \end{pmatrix} \\
0,1,2,3,4,5,6,7,8,9; & & 0,1,2,3,4,5,6,7,8,9; \\
4 \begin{pmatrix} 1^{(0,5)} \\ *^{(*,*)} \\ *^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-5^{0,1,4,5}} \\ \equiv_{0-5^{0,1,4,5}} \end{pmatrix}, & & 4 \begin{pmatrix} 1^{(0,5)} \\ 6^{(2,*)} \\ *^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7^{0-2,8,9}} \end{pmatrix} \\
0,1,2,3,4,5,6,7,8,9; & & 0,1,2,3,4,5,6,7,8,9; \\
4 \begin{pmatrix} 1^{(0,5)} \\ 6^{(2,*)} \\ *^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7^{0-2,8,9}} \end{pmatrix}, & & 4 \begin{pmatrix} 1^{(0,5)} \\ 6^{(2,*)} \\ *^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7^{3,7}} \end{pmatrix} \\
0,1,2,3,4,5,6,7,8,9; & & 0,1,2,3,4,5,6,7,8,9;
\end{array}$$

На примере проекции $P'(4)$ показан еще один из используемых приемов: для второго уровня ветви с пока еще неизвестным корнем (помеченным символом $*$) множество $\bar{M}_{2,*}^*(4)$ вершин, исключенных из числа претендующих на вакансии второго уровня $*$ -й ветви с обычным начертанием, заменено достоверно заполняющими обе $(2,*)$ -вакансии «жирными» вершинами **3, 7** — в соответствии с отмеченными выше свойствами проекций искомого графа все 10 вершин в любой его 2-уровневой проекции обязательно должны быть упомянуты, тогда как эти вершины (третья и седьмая) внесены в перечень запрещенный всех остальных $(l \neq 2, *)$ -вакансий. Поступив аналогично с оставшимися проекциями $P'(v_i) | 4 < v_i \leq 9$ подмножествами $M^*(v_i)$, $\bar{M}^*(v_i)$ и таблицей потенциальной смежности, произведем выбор варианта заполнения вакансий, используя полученные результаты.

В пояснениях к утверждению 1 мы уже указывали, что вершина 5 может быть смежной только вершинам 6 или 7 и вершинам 8 или 9; из таблицы, полученной нами в результате соединения ребром вершин 4 и 5, видим, что вариант соединения ребром вершин 5 и 6 теперь отсутствует, поэтому одна из двух вакансий смежности вершины 5 может быть заполнена только вершиной 7. Ниже приведены проекции и таблица смежности соответствующие введению в граф $G'(V, E')$ ребра 5-7: $E' := E' + (5-7)$.

$$\begin{array}{ccc}
0 \begin{pmatrix} 1^{(4,5)} \\ 2^{(6,7)} \\ 3^{(8,9)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix}, & & 2 \begin{pmatrix} 0^{(1,3)} \\ 6^{(4,*)} \\ 7^{(5,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7} \end{pmatrix} \\
0,1,2,3,4,5,6,7,8,9; & & 0,1,2,3,4,5,6,7,8,9; \\
1 \begin{pmatrix} 0^{(2,3)} \\ 4^{(6,*)} \\ 5^{(7,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7} \end{pmatrix}, & & 3 \begin{pmatrix} 0^{(1,2)} \\ 8^{(*,*)} \\ 9^{(*,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-3,8,9} \\ \equiv_{0-3,8,9} \end{pmatrix} \\
0,1,2,3,4,5,6,7,8,9; & & 0,1,2,3,4,5,6,7,8,9;
\end{array}$$

$$4 \begin{pmatrix} 1^{(0,5)} \\ 6^{(2,*)} \\ *(*,*) \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7,3,7} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$5 \begin{pmatrix} 1^{(0,4)} \\ 7^{(2,*)} \\ *(*,*) \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7,3,6} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$6 \begin{pmatrix} 2^{(0,7)} \\ 4^{(1,*)} \\ *(*,*) \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7,3,5} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$7 \begin{pmatrix} 2^{(0,6)} \\ 5^{(1,*)} \\ *(*,*) \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7,3,4} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$8 \begin{pmatrix} 3^{(0,9)} \\ *(*,*) \\ *(*,*) \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-3,8,9^{0,3,8,9}} \\ \equiv_{0-3,8,9^{0,3,8,9}} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$9 \begin{pmatrix} 3^{(0,8)} \\ *(*,*) \\ *(*,*) \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-3,8,9^{0,3,8,9}} \\ \equiv_{0-3,8,9^{0,3,8,9}} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

0	0	1	2	3	4	5	6	7	8	9
1	0	1	2	3	4	5	6	7	8	9
2	0	1	2	3	4	5	6	7	8	9
3	0	1	2	3	4	5	6	7	8	9
4	0	1	2	3	4	5	6	7	8	9
5	0	1	2	3	4	5	6	7	8	9
6	0	1	2	3	4	5	6	7	8	9
7	0	1	2	3	4	5	6	7	8	9
8	0	1	2	3	4	5	6	7	8	9
9	0	1	2	3	4	5	6	7	8	9

Учитывая утверждение 1 в применении его к вершине (строке таблицы) 4, выбираем соединение ребром вершин 4 и 8 и модифицируем таблицу, проекции и подмножества. В таблице закрываем единственную свободную вакансию вершины 4 (строки 4 таблицы) вершиной 8 и скрываем серым шрифтом «черные» вершины, убирая их из числа соискателей вакансии смежности с вершиной 4 (в данном случае убираем вершину 9). Соответственно, вершиной 4 закрываем одну из двух вакансий вершины 8 и скрываем вершину 4 в строке 9.

$$0 \begin{pmatrix} 1^{(4,5)} \\ 2^{(6,7)} \\ 3^{(8,9)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$1 \begin{pmatrix} 0^{(2,3)} \\ 4^{(6,8)} \\ 5^{(7,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv_9 \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$2 \begin{pmatrix} 0^{(1,3)} \\ 6^{(4,*)} \\ 7^{(5,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-7} \\ \equiv_{0-7} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$3 \begin{pmatrix} 0^{(1,2)} \\ 8^{(4,*)} \\ 9^{(5,*)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv_{0-5,8,9} \\ \equiv_{0-5,8,9} \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$4 \begin{pmatrix} 1^{(0,5)} \\ 6^{(2,*)} \\ 8^{(3,7)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$5 \begin{pmatrix} 1^{(0,4)} \\ 7^{(2,8)} \\ 9^{(3,6)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$6 \begin{pmatrix} 2^{(0,7)} \\ 4^{(1,8)} \\ 9^{(3,5)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$7 \begin{pmatrix} 2^{(0,6)} \\ 5^{(1,9)} \\ 8^{(3,4)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$8 \begin{pmatrix} 3^{(0,9)} \\ 4^{(1,6)} \\ 7^{(2,5)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

$$9 \begin{pmatrix} 3^{(0,8)} \\ 5^{(1,7)} \\ 6^{(2,4)} \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix} \begin{pmatrix} \equiv \\ \equiv \\ \equiv \end{pmatrix},$$

0,1,2,3,4,5,6,7,8,9;

0	0	1	2	3	4	5	6	7	8	9
1	0	1	2	3	4	5	6	7	8	9
2	0	1	2	3	4	5	6	7	8	9
3	0	1	2	3	4	5	6	7	8	9
4	0	1	2	3	4	5	6	7	8	9
5	0	1	2	3	4	5	6	7	8	9
6	0	1	2	3	4	5	6	7	8	9
7	0	1	2	3	4	5	6	7	8	9
8	0	1	2	3	4	5	6	7	8	9
9	0	1	2	3	4	5	6	7	8	9

Модифицируя $P'(1)$, получаем ребро 5-9. Производим те же действия со строками 5, 9 и 8. Продолжая модификацию проекций, дадим дополнительные пояснения к проекции $P'(4)$. Заполнив в ней свободную вакансию первого уровня вершиной 8, обнаруживаем, что второй уровень этой ветви уже заполнен вершинами 3 и 7, что указывает на смежность вершины 8 с этими вершинами, — внесем соответствующее изменение в таблицу смежности искомого графа, оно коснется лишь вершины 7, так как смежность 8 с 3 уже была задана таблицей. Все три вакансии восьмой строки окажутся при этом заполненными, оставшуюся в строке вершину 6 извлекаем из числа потенциально смежных (вводим запрет на ее смежность вершине 8, добавив в подмножество «серых» вершин строки) и делаем то же самое с вершиной 8 в строке 6. В строке 6 при этом остается всего одна «черная» вершина — 9, претендующая на единственную в этой строке вакансию. Заполним эту вакансию вершиной 9 (меняем ее шрифт на полужирный), добавим в выстраиваемый граф ребро (6-9), строка 9 при

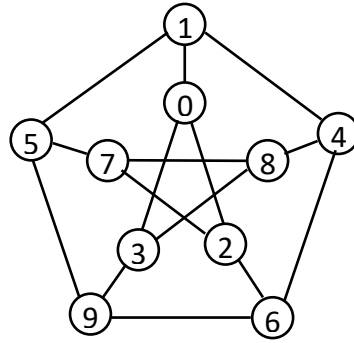
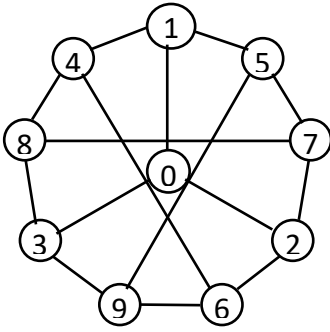
этом также дополняется «жирной» вершиной **6**. При этом таблица потенциальной смежности потеряет статус потенциальной и приобретет окончательный вид таблицы смежности искомого графа: все три вакансии смежности в каждой строке окажутся заполненными в результате проведенных выше действий.

Продолжив модификацию оставшихся проекций с учетом модифицированной на текущий момент таблицы, убедимся в том, что двухуровневые проекции вершинно полны (все вершины из V , помещенные под каждой проекцией станут серыми — будут в ней упомянуты), и полученный таким образом граф полностью соответствует заданным условиям: его порядок $|V| = 10$, степень $s = 3$ и диаметр $d = 2$. Проекции $P'(2)$ и $P'(3)$, предшествующие полученному в проекции $P'(4)$ решению, для большей наглядности оставлены в том виде, в каком они были до рассмотрения $P'(4)$.

Итак, решив представленным выше способом систему проекций исходного графа, представленного в данном случае каркасом искомого графа, мы получили все недостающие для его построения ребра. Каждая из полученных проекций является вершинно, но не реберно полной. Однако, этот граф можно описать всего одной проекцией, для этого достаточно надстроить любую из них еще одним уровнем. Покажем это примерами проекций $P(0)$, $P(3)$ и $P(8)$:

$$\begin{matrix} 0 & \begin{pmatrix} 1^{(4(6,8), 5(7,9))} \\ 2^{(6(4,9), 7(5,8))} \\ 3^{(8(4,7), 9(5,6))} \end{pmatrix}, & 3 & \begin{pmatrix} 0^{(1(4,5), 2(6,7))} \\ 8^{(4(1,6), 7(2,5))} \\ 9^{(5(1,7), 6(2,4))} \end{pmatrix}, & 8 & \begin{pmatrix} 3^{(0(1,2), 9(5,6))} \\ 4^{(1(0,5), 6(2,9))} \\ 7^{(2(0,6), 5(1,9))} \end{pmatrix}.
 \end{matrix}$$

Из приведенных выше надстроенных третьим уровнем проекций можно увидеть, что реплики, впервые появившиеся при этом на уровне 3, имеют оригиналами только вершины уровня 2. Следовательно, длина минимального цикла в каждой из проекций, определяемая суммой номеров уровней оригинальной вершины и ее реплики, равна пяти. Аналогичную картину нетрудно увидеть и из остальных проекций, т. е. обхват нашего 10-вершинного степени 3 графа равен пяти, и этот граф не что иное как всем известная (3,5)- клетка, или граф Петерсена, он же, как обладающий при этих значениях степени и обхвата минимальным порядком, — граф Мура. Изоморфизм полученного нами графа графам, поименованным выше, следует из приведенного ниже рисунка, в котором на общеизвестные изображения последних нанесена нумерация первого.



Приведенный здесь пример генерации предельно $(3,2)$ -компактного графа завершается успешно: система проекций сходится. Однако известно, что такие графы для произвольных сочетаний степени s и диаметра d довольно редки и известны лишь для небольших их значений [24]. Поэтому, за редкими исключениями, система проекций искомого графа с заданными значениями s и d не имеет решения. При этом число вершин, претендующих на вакансии, в одной или в нескольких строках промежуточной таблицы потенциальной смежности становится меньшим числа вакансий, после чего процесс поиска решения можно прекратить, так как такого графа не существует.

При построении $n(s,d)$ -компактного графа с числом вершин n , меньшим предельного, d -уровневые проекции искомого графа в отличие от рассмотренного выше примера, где $c = 0$, содержат реплики (в этом случае

$$c = 1 + s \sum_{i=1}^d (s-1)^{i-1} - n > 0), \text{ что и должно быть учтено свойствами } Q(P'(G)).$$

Дабы не перегружать статью, ограничимся приведением здесь лишь исходных и отдельных полученных из них полных проекций графов с требуемыми свойствами.

Покажем достоверность использования предлагаемого метода для получения графов с диаметром, большим двух. Ниже приведены: слева – исходная вершинно полная 4-уровневая проекция каркаса генерируемого $30(3,4)$ -компактного графа ($s = 3, d = 4$) и справа — также 4-уровневая, но уже полная проекция искомого графа, полученная в результате решения системы из тридцати исходных проекций, описывающих остовное дерево. При построении этого графа автором изначально было задано основополагающее условие, что длина циклов (сумма номеров оригинальных вершин и их реплик) в каждой из проекций не должна быть меньше восьми. Нетрудно убедиться, что полученный граф является $(3,8)$ -клеткой: обхват его равен восьми, а число вершин —

нижнему пределу для четных обхватов, в связи с чем может быть отнесен и к графам Мура [25].

$$0 \left(\begin{array}{l} 1^{(4(10^{(=,=)}, 11^{(=,=)}, 12^{(=,=)}, 13^{(=,=)}),} \\ 2^{(6(14^{(22,23)}, 15^{(24,25)}, 16^{(26,27)}, 17^{(28,29)}),} \\ 3^{(8(18^{(=,=)}, 19^{(=,=)}, 20^{(=,=)}, 21^{(=,=)}),} \end{array} \right) \rightarrow 0 \left(\begin{array}{l} 1^{(4(10^{(22,28)}, 11^{(27,25)}, 12^{(23,29)}, 13^{(26,24)}),} \\ 2^{(6(14^{(22,23)}, 15^{(24,25)}, 16^{(26,27)}, 17^{(28,29)}),} \\ 3^{(8(18^{(23,27)}, 19^{(24,28)}, 20^{(22,26)}, 21^{(25,29)}),} \end{array} \right)$$

Приведем теперь результат использования предложенного формального метода для получения 15(3,3)-компактного графа из исходного графа, отличного от применяемых выше остовных деревьев. Предположим, нам требуется построить ВС с централизованным управлением, устойчивым к отказам кратности два. Для создания такого отказоустойчивого центрального ядра управления, исходный граф здесь образован 3-циклом с вершинами 0, 1, 2; затем к вершинам ядра добавим по две смежные вершины (соответственно, 3 и 4, 5 и 6, 7 и 8), завершив окружения каждой из этих шести вершин оставшимися (с девятой по четырнадцатую) вершинами.

$$0 \left(\begin{array}{l} 1^{(2(5,6)},} \\ 2^{(1(7,8)},} \\ 3^{(9(=,=)},} \\ 4^{(10(=,=)},} \end{array} \right) \rightarrow 0 \left(\begin{array}{l} 1^{(2(0,7,8)}, 5^{(10,11,14)}, 6^{(9,12,13)},} \\ 2^{(1(0,5,6)}, 7^{(9,10,13)}, 8^{(11,12,14)},} \\ 3^{(9(6,7,11)}, 12^{(6,8,10)}, 14^{(5,8,13)},} \\ 4^{(10(5,7,12)}, 11^{(5,8,9)}, 13^{(6,7,14)},} \end{array} \right)$$

3. Заключение

В работе впервые описаны формальные процедуры детерминированного получения графов с заданными свойствами. Основой метода являются оригинальное их описание с помощью проекций, свойства которых биективны свойствам искомого графа. Решение системы проекций в отношении этих свойств позволяет либо получить проективное описание такого графа, либо, при выявлении ее несовместности, доказать невозможность его существования. Биекция свойств оптимальных топологий интерконнекта вычислительных систем и свойств генерируемых графов позволяет использовать предлагаемую методику в заполнении ниши теоретических основ *детерминированного и реальновременного* проектирования и использования крупномасштабных вычислительных систем и суперкомпьютеров. Именно это, прежде всего, и явилось предпосылкой разработки как оригинального метода проективного описания графов, так и его последующего развития и использования, включая предлагаемую в данной работе методику.

Проблема построения графов с заданными свойствами является чрезвычайно насущной в теории графов. Еще в 1981 году Вонг в статье

[26] отметил: «Регулярный граф с заданными обхватом и валентностью получить довольно сложно. Особенно, когда обхват или валентность графа велики. И только с помощью компьютера, *некоторые* из таких графов могут быть успешно получены». С тех пор в отношении формальных методов синтеза графов с заданными свойствами практически мало что изменилось. Автор надеется, что использование предложенного метода не только восполнит эти пробелы в теории графов, но позволит производить менее трудоемкий и более качественный систематический анализ и сопоставление исследуемых графов. Естественным образом это скажется также и на всех приложениях, использующих аппарат теории графов.

Благодарности: выражаю искреннюю признательность академикам РАН Александру Леонидовичу Асееву и Александру Васильевичу Латышеву, благодаря которым имею возможность продолжать исследования по этой тематике.

СПИСОК ЛИТЕРАТУРЫ

1. *Задорожный А.Ф., Мелентьев В.А.* О совместимости топологий параллельных задач и систем // 13-я мультиконференция по проблемам управления, материалы конференции «Информационные технологии в управлении» (ИТУ-2020). – 2020. – СПб.: АО «Концерн «ЦНИИ «Электроприбор». С. 162–164.
2. *Мелентьев В.А.* Компактные структуры вычислительных систем и их синтез / Управление большими системами. – 2011. – Вып. 32. М.: ИПУ РАН. – С. 241–261.
3. *Мелентьев В.А.* О топологической компактности вычислительных систем / ISJ Theoretical & Applied Science. – 2014. – 11 (19), – С. 59–65. DOI: 10.15863/TAS.2014.11.19.12
4. *Melentiev V.A.* On scalability of computing systems with compact topology / Theoretical & Applied Science. – 2016. – 11 (43). – С. 164–169. DOI: 10.15863/TAS.2016.11.43.30
5. *Раннопорт А. М.* Метрические характеристики графов сетей коммуникаций // Труды Института системного анализа Российской академии наук. – 2005. – Т. 14. – С. 141–147.
6. *Мелентьев В. А.* О топологической масштабируемости вычислительных систем / Управление большими системами. – 2015. – Вып. 58. М.: ИПУ РАН. – С.115–143.

7. *Мелентьев В. А., Шубин В.И., Задорожный А.Ф.* Топологическая масштабируемость гиперкубических параллельных систем и задач // Theoretical & Applied Science. – 2015. – 11(31), С. 122–129. DOI: 10.15863/TAS.2015.11.31.19
8. *Melentiev V.A.* The bracket form of graph description and its use in structural investigation of robust computer systems // Optoelectronics Instrumentation and Data Processing. – 2000. – 4. – P. 34–47.
9. *Melent'ev V.A.* The bracket Pattern of a Graph // Proceedings of the 6-th International Conference on Pattern Recognition and Image Analysis: New Information Technologies, PRIA-6-2002, October 21-26 2002, Velikiy Novgorod, Russian Federation. – 2002. – P. 57–61.
10. *Мелентьев В. А.* Формальные основы скобочных образов в теории графов // Труды Второй Международной конференции Параллельные вычисления и задачи управления РАСО'2004, 4-6 октября 2004: М., ИПУ РАН. – 2004. – С. 694–706.
11. *Мелентьев, В. А.* Формальный подход к исследованию структур вычислительных систем. // Вестник Томского госуниверситета. – 2005. Приложение 14. – С. 167–172.
12. *Мелентьев, В. А.* Аналитический подход к синтезу регулярных графов с заданными значениями порядка, степени и обхвата // Прикл. дискр. мат. – 2010. – 2(8). – С. 74–86.
13. *В. А. Мелентьев* Аналитический подход к синтезу регулярных структур отказоустойчивых систем // Прикл. дискр. мат. – 2010, Приложение 3. – С. 74–75.
14. *В. А. Мелентьев* Метрика, цикломатика и синтез топологии систем и сетей связи // Труды шестой Международной конференции «Параллельные вычисления и задачи управления РАСО'2012, Москва, 24–26 октября 2012». – 2012. – С. 10–25.
15. *Корнюшко, В. Ф., Панов, А. В., Богунова, И. В., Николаева, О. М., & Флид, А. А.* Системный подход к информационной поддержке фармацевтической разработки готовых лекарственных средств // Тонкие химические технологии. – 2018. – Т. 13. – №. 2. – С. 91–99/
16. *Елисеев А. И., Минин Ю. В., Мартельянов Ю. Ф.* К вопросу об исследовании графов сетевых информационных систем на толерантность в отношении достижимости вершин // Вестник Воронежского института ФСИН России. – 2011. – №. 2. – С. 55–57.
17. *Volkova A.* A Technical Translation of Melentiev's Graph Representation Method with Commentary. – 2018. – University Honors Theses. Paper 503. DOI: 10.15760/honors.507.

18. Мелентьев В. А. Использование метода Мелентьева представления графов для выявления клик и анализа топологий вычислительных систем // Theoretical & Applied Science. – 2018. –12(68). – С. 201–211.
19. Мелентьев В. А. Использование метода Мелентьева представления графов для выявления и перечисления циклов заданной длины // Theoretical & Applied Science. – 2018. –11(67). – С. 85–91.
20. Мелентьев В. А. Проблемы изоморфизма и толерантности графов в теории отказоустойчивости систем // Труды IV Международной конференции «Идентификация систем и задачи управления» SICPRO'05. Москва, ИПУ РАН. – 2005. – С. 28–30.
21. Мелентьев В. А. Авторский подход к топологическому моделированию параллельных вычислительных систем. – 2020 – PREPRINTS.RU. DOI: 10.24108/preprints-3112009.
22. Melent'ev V.A. Author's approach to topological modeling of parallel computation systems. // Journal of Mechanics of Continua and Mathematical Sciences – 2020 – Special Issue. 8. – P. 224–237.
23. В. А. Мелентьев Компактные графы и детерминированный алгоритм их синтеза // Прикл. дискр. мат. – 2011. – Приложение 4. – С. 94–96.
24. P. Wong Cages — a survey // Journal of Graph Theory. – 1982. – Т. 6. – № 1. – С. 1–22.
25. Dégila J. R., Sanso B. A survey of topologies and performance measures for large-scale networks // IEEE Communications Surveys & Tutorials. – 2004. – Т. 6. – № 4. – С. 18–31.
26. Wong P. K. On the smallest graphs of girth 10 and valency 3 // Discrete Mathematics. – 1983. – Т. 43. – № 1. – С. 119–124.

REFERENCES

1. Zadorozhny A.F. and Melent'ev V.A. On the compatibility of the topologies of parallel tasks and computing systems // 13-ya mul'tikonferenciya po problemam upravleniya, materialy konferencii «Informacionnye tekhnologii v upravlenii» (ITU-2020). – 2020. – SPb.: AO «Koncern «CNIИ «Elektropribor», 2020. S. 162–164. (In Russ.).
2. Melent'ev V.A. Compact structures of computer systems and their synthesis // Upravlenie bol'shimi sistemami. – 2011. – 32. – P. 241–261. (In Russ.).
3. Melent'ev V.A. About topological compactness of computing systems // ISJ Theoretical & Applied Science. – 2014. – 11 (19). – P. 59–65. (In Russ.). DOI: 10.15863/TAS.2014.11.19.12
4. Melent'ev V.A. On scalability of computing systems with compact topology // Theoretical & Applied Science. – 2016. – 11 (43). – P. 164–169. (In Russ.). DOI: 10.15863/TAS.2016.11.43.30

5. *Rappoport A.M.* Metriccheskie kharakteristiki graphov setej kommunikacij // Trudy Instituta systemnogo analiza Rossijskoj akademii nauk. – 2005. – T. 14. – S. 141–147. (In Russ.).

6. *Melent'ev V.A.* On topological scalability of computing systems // Upravlenie bol'shimi systemami. – 2015. – Issue 58. – P. 115–143. (In Russ.).

7. *Melent'ev V.A., Shubin V.I., Zadorozhny A.F.* Topological scalability of hypercubic parallel systems and tasks // Theoretical & Applied Science. – 2015. – 11(31). – P. 122–129. (In Russ.). DOI: 10.15863/TAS.2015.11.31.19

8. *Melentiev V.A.* The bracket form of graph description and its use in structural investigation of robust computer systems // Optoelectronics Instrumentation and Data Processing. – 2000. – 4. – P. 34–47.

9. *Melent'ev V.A.* The bracket Pattern of a Graph // Proceedings of the 6th International Conference on Pattern Recognition and Image Analysis: New Information Technologies, PRIA-6-2002, October 21-26. – 2002. – Velikiy Novgorod, Russian Federation. – P. 57–61.

10. *Melent'ev V.A.* The formal rudiments of a bracket patterns' in graph theory // Proceedings of the II International Conference “Parallel Computation and Control Problems” PACO'2004, October 4-6 2004: Moscow, Russia. ICS RAS. – P. 694–706. (In Russ.).

11. *Melent'ev V.A.* Formal'nyj podhod k issledovaniyu struktur vychislitel'nyh system // Vestnik Tomskogo gosuniversiteta. – 2005. – Prilozhenie 14. – S. 167–172. (In Russ.).

12. *Melent'ev V.A.* An analytical approach to the synthesis of regular graphs with preset values of the order, degree and girth // Prikl. Diskr. Mat. – 2010. – 2(8). – P. 74–86. (In Russ.).

13. *Melent'ev V.A.* An analytical approach to the synthesis of regular structures of fault-tolerant systems // Prikl. Diskr. Mat. – 2010. – supplement 3. – P. 74–75. (In Russ.).

14. *Melent'ev V.A.* The metric, cyclomatic and synthesis of topology of systems and networks // Proceedings of the Sixth International Conference “Parallel Computation and Control Problems” PACO'2012, October 24–26 2012: Moscow, Russia. ICS RAS. – 2012. – P. 10–25. (In Russ.).

15. *Kornushko V.F., Panov A.V., Bogunova I.V., Nikolayeva O.M., Flid A.A.* System approach to informational support of pharmaceutical development of finished medicinal products // Fine Chemical Technologies. – 2018. – 13(2). – P. 91–99. (In Russ.).

16. *Eliseev A.I., Minin Yu.V., Martemjyanov Ju.F.* Tolerance research of graphs of network information systems in case of the reaching its vertexes // Vestnik of Voronezh Institute of the Russian Federal Penitentiary Service. – 2011. – 2. – P. 55–57. (In Russ.).

17. *Volkova A.* A Technical Translation of Melentiev's Graph Representation Method with Commentary. – 2018. – University Honors Theses. Paper 503. DOI: 10.15760/honors.507.

18. *Melent'ev V.A.* Use of Melentiev's graph representation method for detection of cliques and the analysis of topologies of computing systems // Theoretical & Applied Science. – 2018. – 12(68). – P. 201–211. (In Russ.).

19. *Melent'ev V.A.* Use of Melentiev's graph representation method for identification and enumeration of circuits of the given length // Theoretical & Applied Science. – 2018. – 11(67). – P. 85–91. (In Russ.).

20. *Melent'ev V.A.* Problems of isomorphism and tolerance of graphs in the systems' fault-tolerance theory // Proceedings of the IV International Conference System Identification and Control Problems SICPRO'05., 25-28 January 2005.: Institute of Control Sciences, Moscow, Russia. – 2005, – P. 532–549. (In Russ.).

21. *Melent'ev V.A.* Author's approach to topological modeling of parallel computation systems. – 2020. DOI: 10.24108/preprints-3112009. (In Russ.).

22. *Melentiev V.A.* Author's approach to topological modeling of parallel computation systems. // Journal of Mechanics of Continua and Mathematical Sciences – 2020 – Spl. Is., No. 8. – P. 224–237.

23. *Melent'ev V.A.* Compact graphs and the deterministic algorithm for their synthesis // Prikl. Diskr. Mat., – 2011. – Supplement 4. – P. 94–96. (In Russ.).

24. *Wong P.* Cages — a survey // Journal of Graph Theory. – 1982. – V. 6, № 1. – P. 1–22.

25. *Melent'ev V.A.* Izomorfizm grafov i ih obrazov v issledovaniyah otkazoustojchivosti system // Vestnik Tomskogo gosuniversiteta. – 2005. – Prilozhenie 14. – S. 182–190. (In Russ.).

26. *Wong P. K.* On the smallest graphs of girth 10 and valency 3 // Discrete Mathematics. – 1983. – T. 43. – №. 1. – C. 119–124.