



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

G06Q 10/06 (2021.02); G06F 9/5038 (2021.02); G06F 9/4887 (2021.02)

(21)(22) Заявка: 2020130033, 11.09.2020

(24) Дата начала отсчета срока действия патента:
11.09.2020

Дата регистрации:
13.07.2021

Приоритет(ы):

(22) Дата подачи заявки: 11.09.2020

(45) Опубликовано: 13.07.2021 Бюл. № 20

Адрес для переписки:

119234, Москва, Ломоносовский проспект 27,
строение 1, Московский государственный
университет имени М.В. Ломоносова, Фонд
"Национальное интеллектуальное развитие"

(72) Автор(ы):

Смелянский Руслан Леонидович (RU),
Антоненко Виталий Александрович (RU),
Чупахин Андрей Андреевич (RU),
Колосов Алексей Михайлович (RU)

(73) Патентообладатель(и):

Федеральное государственное бюджетное
образовательное учреждение высшего
образования «Московский государственный
университет имени М.В.Ломоносова» (МГУ)
(RU)

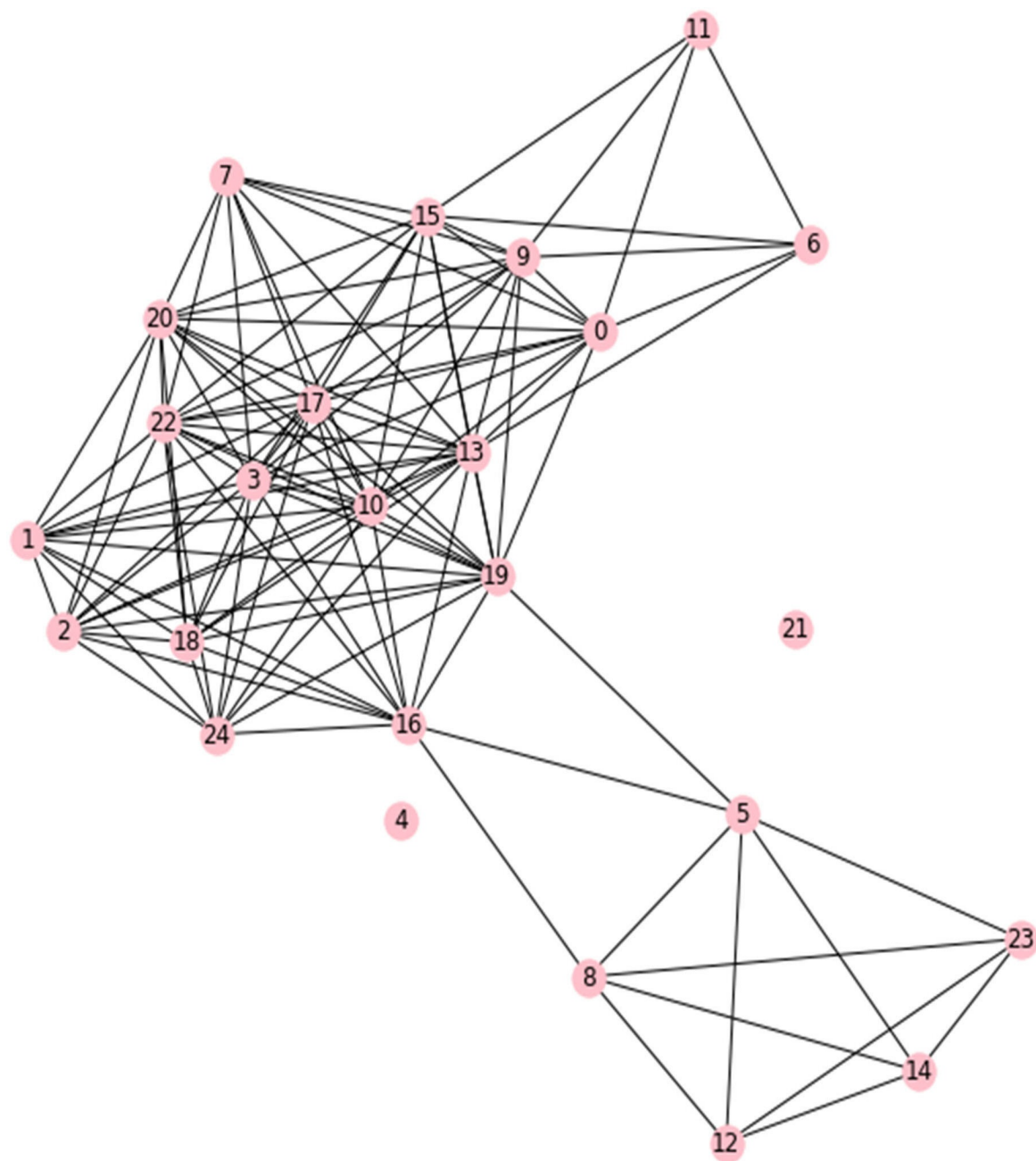
(56) Список документов, цитированных в отчете
о поиске: US 8225119 В, 17.07.2012. US 2018/
0060132 А, 01.03.2018. CN 107729338 А,
23.02.2018. RU 2660614 С2, 06.07.2018. RU
2630753 С2, 12.09.2017.

(54) СПОСОБ ФОРМИРОВАНИЯ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА

(57) Реферат:

Предложенное изобретение относится к области компьютерных технологий. Технический результат, достигаемый при использовании предложенного изобретения, заключается в обеспечении оптимального покрытия типов решаемых задач с помощью минимального количества вычислителей за счет, в том числе, устранения дублирования вычислителей в составе вычислительного комплекса, что обеспечивает формирование вычислительного комплекса оптимальной полноты. Способ формирования вычислительного комплекса оптимального состава включает следующие этапы:

формирование множества вычислителей, являющихся кандидатами в состав вычислительного комплекса, серийное выполнение на каждом вычислителе набора бенчмарков с определением и фиксацией времени выполнения каждого бенчмарка из набора, составление всех возможных пар вычислителей и определение модуля попарного коэффициента корреляции Пирсона (ккП) для каждой пары, формирование вычислительного комплекса с исключением повторных включений одного и того же вычислителя в комплекс. 4 з.п. ф-лы, 2 ил., 5 табл.



Фиг. 1



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) ABSTRACT OF INVENTION

(52) CPC

G06Q 10/06 (2021.02); G06F 9/5038 (2021.02); G06F 9/4887 (2021.02)

(21)(22) Application: 2020130033, 11.09.2020

(24) Effective date for property rights:
11.09.2020Registration date:
13.07.2021

Priority:

(22) Date of filing: 11.09.2020

(45) Date of publication: 13.07.2021 Bull. № 20

Mail address:

119234, Moskva, Lomonosovskij prospekt 27,
stroenie 1, Moskovskij gosudarstvennyj universitet
imeni M.V. Lomonosova, Fond "Natsionalnoe
intellektualnoe razvitie"

(72) Inventor(s):

Smelyanskij Ruslan Leonidovich (RU),
Antonenko Vitalij Aleksandrovich (RU),
Chupakhin Andrej Andreevich (RU),
Kolosoov Aleksej Mikhajlovich (RU)

(73) Proprietor(s):

Federalnoe gosudarstvennoe byudzhethoe
obrazovatelnoe uchrezhdenie vysshego
obrazovaniya «Moskovskij gosudarstvennyj
universitet imeni M.V.Lomonosova» (MGU)
(RU)

(54) METHOD FOR FORMING COMPUTER COMPLEX

(57) Abstract:

FIELD: computer technology.

SUBSTANCE: proposed invention relates to the field of computer technology. The method of forming a computational complex of the optimal composition includes the following stages: formation of a set of calculators that are candidates for the composition of a computer complex, serial execution of a set of benchmarks on each computer with the determination and fixation of the execution time of each benchmark from the set, compilation of all possible pairs of calculators and determination of the modulus of the

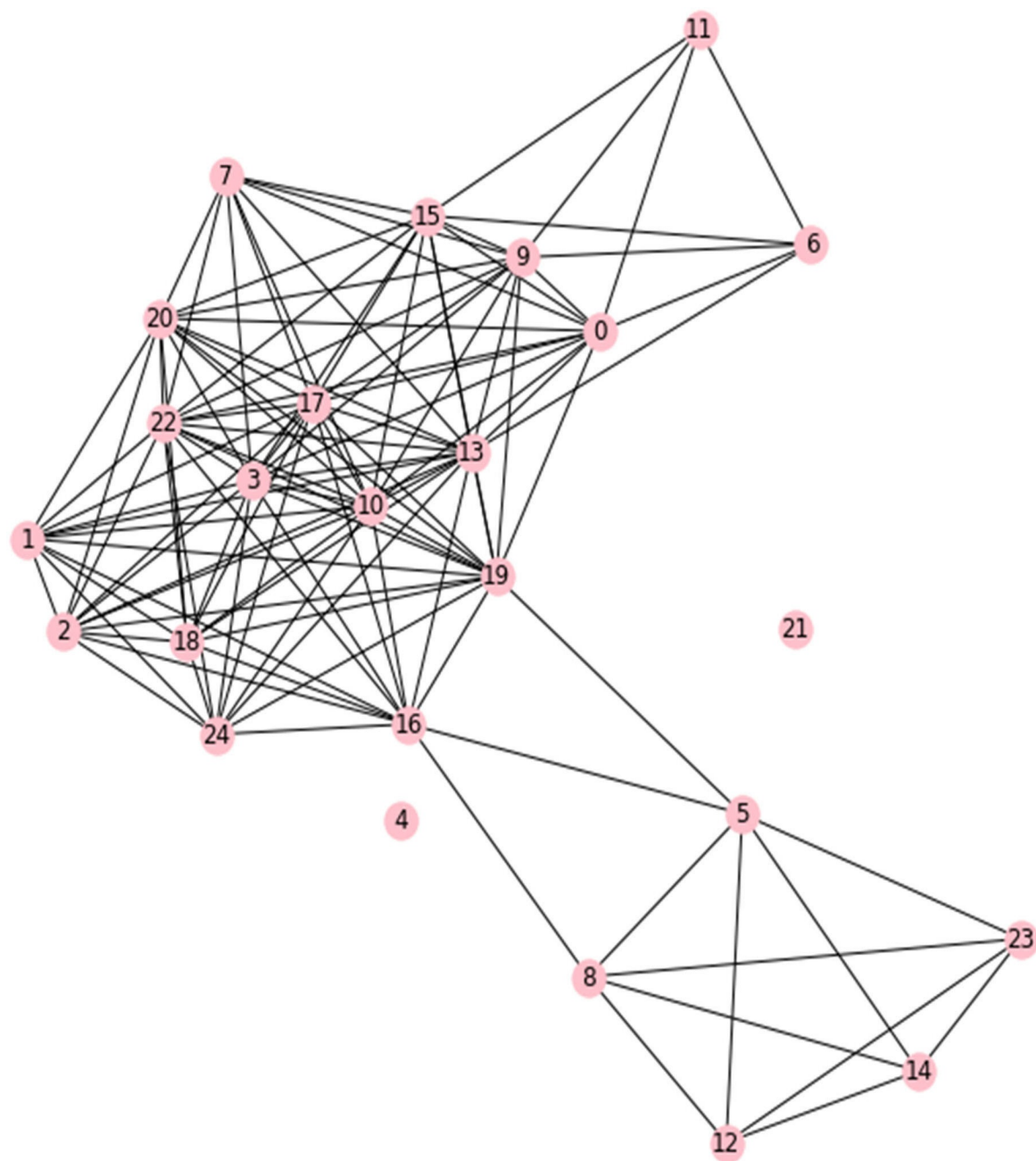
pairwise Pearson correlation coefficient (PCC) for each pair, the formation of a computing complex with the exclusion of repeated connections of the same calculator in the complex.

EFFECT: invention makes it possible to provide optimal coverage of the types of problems to be solved using a minimum number of computers due to, inter alia, eliminating duplication of computers in the computing complex, which ensures the formation of a computing complex of optimal completeness.

5 cl, 2 dwg, 5 tbl

RU 2 751 441 C1

RU 2 751 441 C1



Фиг. 1

Область техники, к которой относится изобретение

Заявляемое изобретение относится к области компьютерных технологий, а именно, к сетевым компьютерным технологиям, позволяющим объединять и использовать вычислительные ресурсы различных видов для проведения междисциплинарных исследований, например, в области биоинформатики, биоинженерии, космических исследований, в медицине, физике и др. Разные направления исследования зачастую требуют применения вычислителей различных типов, в результате чего задача объединения вычислительных ресурсов в единую систему, работающую над решением одной задачи, становится чрезвычайно актуальной. Выбор вычислителей для формирования такой системы представляет собой нетривиальный процесс, требующий использования специализированных аппаратно-программных решений.

Уровень техники

На данном этапе развитие многих высокотехнологичных отраслей науки и техники подразумевает использование высокопроизводительных систем обработки данных.

Для таких задач используют модульные вычислительные системы, состоящие из отдельных серверов, вычислительные ресурсы которых объединены в единое целое посредством специального программного обеспечения. Благодаря специально разработанным диспетчерским системам пакетной обработки данных, каждая задача может быть распределена на все узлы объединения, в результате чего для решения каждой задачи могут быть использованы ресурсы всей системы. Существенное влияние на общую производительность и эффективность использования такого объединения вычислительных ресурсов (вычислительного комплекса) оказывает принцип объединения вычислителей в комплекс, а также выбранный подход к реализации каналов межузлового взаимодействия.

В качестве вычислителей, объединяемых в такие вычислительные системы, могут быть использованы, например, персональные компьютеры, рабочие станции, серверы, мэйнфреймы, кластеры, представляющие собой распределённые системы компьютеров, суперкомпьютеры или облачные ИТ-ресурсы.

Известные из уровня техники способы формирования вычислительных комплексов (объединения вычислителей) *имеют целью*, в основном, обеспечение снижения энергозатрат или экономическое преимущество.

Так, например, из публикации US9342376B2 «Способ, система и устройство для динамического энергоэффективного планирования заданий в среде облачных вычислений» («Method, system, and device for dynamic energy efficient job scheduling in a cloud computing environment», Intel Corp - <https://patents.google.com/patent/US9342376B2>) известны способ, система и устройство для энергосберегающего планирования заданий в вычислительной среде центра обработки данных, включающего в себя главный узел. Главный узел может периодически принимать энергетические данные от подчиненных узлов и динамически назначать вычислительные задачи, которые должны выполняться подчиненными узлами, на основе энергетических данных. В качестве вычислителей в известном техническом решении используют вычислительные узлы, а в качестве вычислительного комплекса - центр обработки данных.

Из публикации US20100111105, «Центр обработки данных и дизайн центра данных» («Data center and data center design», Hewlett Packard Enterprise Development LP - <https://patents.google.com/patent/US20100111105>) известен центр обработки данных, содержащий: множество секций центра обработки данных, причем каждая секция имеет различный предварительно определенный уровень надежности; и множество наборов приложений, причем каждый набор приложений заполняется на одном из множества участков центра

обработки данных. В качестве вычислителей в известном техническом решении используют секции с разным уровнем надёжности (разные уровни надёжности имеют разную стоимость), а в качестве вычислительного комплекса - центр обработки данных.

Как отмечено выше, известные способы основаны на применении отборочного критерия, не обеспечивающего формирования вычислительного комплекса, имеющего оптимальную загрузку, как по времени выполнения отдельных задач, так и по затрачиваемым ресурсам.

Наиболее близким по технической сущности к заявляемому способу является способ определения состава вычислительного комплекса, обеспечивающий оптимальное энергопотребление набора вычислителей (US8225119, Energy-aware server management, Microsoft Technology Licensing LLC, <https://patents.google.com/patent/US8225119>). При таком подходе производится управление энергопотреблением в ферме серверов путем переключения отдельных серверов между активным и неактивным состояниями, с сохранением времени отклика для фермы серверов на предварительно определенном уровне. Для этого при реализации способа в зависимости от ожидаемой нагрузки определяли, какие вычислители должны быть включены (остальные вычислители при этом выключены). Таким образом, производили выбор подмножества вычислителей, определяющих состав вычислительного комплекса для состояния системы в определенный момент времени. Для определения состава вычислительного комплекса прогнозируют будущую рабочую нагрузку для набора серверов, включаемых в вычислительный комплекс. Отдельные серверы имеют, по крайней мере, два энергетических состояния: включено и выключено. Энергетические состояния отдельных серверов корректируются в зависимости от прогнозируемой будущей рабочей нагрузки.

Недостатком известного способа является тот факт, что такой подход не обеспечивает формирование вычислительного комплекса, обеспечивающего максимальное покрытие типов решаемых задач с помощью минимального количества вычислителей, то есть формирование вычислительного комплекса максимальной полноты или производительности.

В общем случае, для каждой программы в отдельности наиболее оптимальным способом ее исполнения является направление её для выполнения на вычислитель, обеспечивающий наиболее быстрое исполнение программы данного типа, без учета характеристики временных классов вычислителей, в результате чего состав вычислительного комплекса в известном решении формируют различные представители всех временных классов, что может приводить к избыточному включению в комплекс вычислителей одних и тех же временных классов, то есть к избыточному количеству вычислителей в вычислительном комплексе в целом, что соответствует понятию «недостаточности полноты вычислительного комплекса», а также снижению эффективности инвестиций в её функционирование в дальнейшем. В результате, известный способ формирования вычислительного комплекса не обеспечивает формирование оптимального состава вычислительного комплекса с точки зрения максимального покрытия типов решаемых задач (максимально возможного количества выполняемых программ разных типов) при минимальном количестве включенных в комплекс вычислителей.

Таким образом, известные вычислительные комплексы не обеспечивают возможности группировки входящих в них вычислителей по временным классам, в результате чего отсутствует возможность приоритетного выполнения той или иной программы вычислителем соответствующего временного класса, то есть все вычислители всех временных классов имеют равный приоритет для задач любого типа.

Между тем, следует отметить, что более рациональным представляется подход, когда для выполнения программы используют вычислители определенных временных классов.

Временной класс - последовательность временных интервалов фиксированного размера, задающая измерительную шкалу времени выполнения программы на

5 *вычислителе. Размер интервала является единицей измерения времени выполнения программы на вычислителе в данном временном классе.* Временные классы могут формировать интервалы времени, например, 10 секунд (тогда измерительная шкала данного временного класса выглядит как 0, 10, 20, 30 и т.д.) или 15 секунд (тогда измерительная шкала данного временного класса выглядит как 0, 15, 30, 45 и т.д.). По

10 аналогии временные классы могут быть определены и для других интервалов времени. Соответственно, если программа выполняется на определенном вычислителе за 40 секунд, то полагают, что данная программа относится к первому временному классу вычислителей (где измерительная шкала сформирована как 0, 10, 20, 30 и т.д.) и не относится ко второму (где измерительная шкала сформирована как 0, 15, 30, 45 и т.д.).

15 Таким образом, на каждом вычислителе вычислительного комплекса возможно осуществить запуск программы любого временного класса. При этом выполненные тем или иным вычислителем программы не соотносятся с определенным временным классом, поскольку *после* получения результатов выполнения программы информация о временном классе программы становится излишней, так как её повторные запуски

20 маловероятны. Между тем на эффективности использования вычислительного комплекса несвязанность программы при ее запуске с определенным временным классом вычислителя сказывается, поскольку при отсутствии корреляции программ временным классам вычислителей, могут быть выявлены простои отдельных вычислителей или, наоборот, слишком длительное выполнение программы при возможности ее более

25 быстрого завершения на другом вычислителе. При отсутствии такой группировки вычислителей по временным классам выполняемых программ также может уменьшиться полнота вычислительного комплекса, поскольку не будут исключены вычислители, дублирующие одинаковые временные классы.

Техническая проблема, решаемая посредством заявляемого изобретения, заключается

30 в необходимости преодоления недостатков, присущих известным способам формирования вычислительного комплекса, за счет создания способа определения такого состава вычислительного комплекса, который позволит обеспечить формирование вычислительного комплекса, обеспечивающего максимальное покрытие классов решаемых задач с помощью минимального количества вычислителей.

35 Краткое раскрытие сущности изобретения

Технический результат, достигаемый при использовании заявляемого изобретения, заключается в обеспечении оптимального покрытия типов решаемых задач с помощью минимального количества вычислителей за счет, в том числе, устранения дублирования вычислителей в составе вычислительного комплекса, что обеспечивает формирование

40 вычислительного комплекса оптимальной полноты.

Количество выбранных для формирования вычислительного комплекса вычислителей в соответствии с заявляемым способом является минимально возможным (необходимым и достаточным) для решения задач разной степени сложности, при этом вычислительный комплекс формируется таким образом, что в результате применения характеристики

45 временного класса не допускается простоя или перегрузки отдельных вычислителей комплекса, что способствует экономии как потребляемых энергоресурсов, так и затрачиваемых мощностей и машинного времени.

Заявляемый способ обеспечивает выбор подмножества вычислителей из множества

вычислителей таким образом, что выбирается состав вычислительного комплекса, оптимальный по соотношению числа участников вычислительного комплекса к количеству различных временных классов, т.е. обеспечивается максимальное покрытие временных классов минимальным количеством вычислителей.

5 Заявляемый технический результат достигается тем, что способ формирования вычислительного комплекса оптимального состава включает следующие этапы:

1) формирование множества вычислителей, являющихся кандидатами в состав вычислительного комплекса,

10 2) формирование набора бенчмарков, предназначенных к выполнению на каждом вычислителе из множества вычислителей, сформированного на этапе 1),

3) серийное выполнение на каждом вычислителе набора бенчмарков с определением и фиксацией времени выполнения каждого бенчмарка из набора,

4) определение среднего времени выполнения в серии каждого бенчмарка отдельно и всего набора бенчмарков в целом на каждом вычислителе из множества вычислителей,

15 5) составление всех возможных пар вычислителей из множества, сформированного на шаге 1) и определение модуля попарного коэффициента корреляции Пирсона (ккП) для каждой пары на основании данных, полученных на этапах 3) и 4),

для каждого из полученных значений модуля (ккП) из диапазона от 0,1 до 1 с шагом от 0,01 до 0,05 выполняют этапы 6)-9):

20 6) выделение групп вычислителей, включающих пары вычислителей, имеющих значение модуля попарного коэффициента корреляции Пирсона (ккП) больше заданного значения,

7) определение множества вычислителей из вычислителей, вошедших в группы, сформированные на шаге 6), включающего по одному вычислителю из всех групп,

25 выделенных на шаге 6),

8) формирование вычислительного комплекса из вычислителей, определенных на шаге 7), с исключением повторных включений одного и того же вычислителя в комплекс,

9) определение полноты сформированного вычислительного комплекса, для чего для каждого сформированного на шаге 8) вычислительного комплекса определяют отношение количества групп, полученных на шаге 6), к количеству вычислителей, составляющих множество на шаге 1),

30 10) выбирают наиболее оптимальный вычислительный комплекс по наибольшему значению его полноты, определенной на этапе 9).

При формировании набора бенчмарков для выполнения этапа 2) выбирают

35 бенчмарки в количестве не менее 5 штук: а) принадлежащие различным временным классам, то есть бенчмарки, время выполнения которых на одном и том же вычислителе находится в разных измерительных временных шкалах, б) характеризующиеся минимальным и максимальным временем выполнения, в) характеризующиеся различными значениями требуемых машинных ресурсов для выполнения.

40 Серийное выполнение набора бенчмарков на каждом вычислителе включает, по меньшей мере, три последовательных запуска набора. На этапе 6) для формирования групп вычислителей используют алгоритм формирования клик. В качестве такого алгоритма может быть использован алгоритм Брона-Кербоша.

Краткое описание чертежей

45 Заявляемое изобретение поясняется следующими чертежами, где:

На фиг. 1 схематично представлена часть графа, вершины которого представляют собой вычислители, являющиеся кандидатами в создаваемый вычислительный комплекс, а рёбрами соединены вычислители, ккП между которыми превышает заранее заданное

пороговое значение (в примере равное 0.75).

На фиг. 2 представлен график зависимости полноты формируемого вычислительного комплекса от порогового значения ккП при минимальном размере группы, равном трём.

5 Используемая терминология

Для наилучшего понимания заявляемой сущности изобретения ниже приведены основные термины и определения, используемые в дальнейшем описании.

Вычислительный комплекс - вычислительная среда, в которой все участники могут делиться и/или использовать вычислительные ресурсы друг друга.

10 *Вычислитель* - участник вычислительного комплекса.

Кандидат в состав вычислительного комплекса - потенциальный участник вычислительного комплекса.

Бенчмарк - эталонный тест производительности компьютерной системы (программа).

15 *Коэффициент корреляции Пирсона* [http://www.machinelearning.ru/wiki/index.php?title=Коэффициент_корреляции_Пирсона] характеризует существование линейной зависимости между двумя величинами.

Пусть даны две выборки размера m : $x^m = (x_1, \dots, x_m)$, $y^m = (y_1, \dots, y_m)$, коэффициент корреляции Пирсона рассчитывается по формуле:

$$20 \quad r_{xy} = \frac{\sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^m (x_i - \bar{x})^2 \sum_{i=1}^m (y_i - \bar{y})^2}}$$

25 где $\bar{x}, \bar{y}$ - выборочные средние x^m и y^m , $r_{xy} \in [-1, 1]$.

Коэффициент корреляции Пирсона называют также теснотой линейной связи:

$|r_{xy}| = 1 \Rightarrow x, y$ линейно зависимы,

$r_{xy} = 0 \Rightarrow x, y$ линейно независимы.

30 *Алгоритм формирования клик* - алгоритм, определяющий в неориентированном графе все максимальные по включению полные подграфы.

Таких алгоритмов известно несколько, возможен выбор любого из них, на сущность изобретения это не влияет. Одним из самых эффективных алгоритмов формирования клик является алгоритм Брона-Кербоша (https://ru.wikipedia.org/wiki/Алгоритм_Брона_Кербоша).

35 Алгоритм оперирует тремя множествами вершин графа:

- Множество *compsub* - множество, содержащее на каждом шаге рекурсии полный подграф для данного шага. Строится рекурсивно.
- Множество *candidates* - множество вершин, которые могут увеличить *compsub*
- Множество *not* - множество вершин, которые уже использовались для расширения

40 *compsub* на предыдущих шагах алгоритма.

Алгоритм является рекурсивной процедурой, применяемой к этим трем множествам.

ПРОЦЕДУРА *extend(candidates, not)*:

ПОКА *candidates* НЕ пусто И *not* НЕ содержит вершины, СОЕДИНЕННОЙ СО ВСЕМИ вершинами из *candidates*,

45 ВЫПОЛНЯТЬ:

1 Выбираем вершину v из *candidates* и добавляем её в *compsub*

2 Формируем *new_candidates* и *new_not*, удаляя из *candidates* и *not* вершины, не СОЕДИНЕННЫЕ с v

3 ЕСЛИ *new_candidates* и *new_not* пусты

4 ТО *compsub* - клика

5 ИНАЧЕ рекурсивно вызываем *extend(new_candidates, new_not)*

6 Удаляем *v* из *compsub* и *candidates*, и помещаем в *not*

5 *Алгоритм формирования состава вычислительного комплекса* - алгоритм, определяющий минимальный состав представителей заданного набора множеств (минимальное покрытие).

Таких алгоритмов может быть предложено несколько, возможен выбор любого из них, на сущность изобретения выбор не влияет. Одним из самых простых алгоритмов является жадный алгоритм.

Алгоритм оперирует двумя множествами и словарём:

- Множество *memb* - множество, содержащее текущий состав вычислительного комплекса для данной итерации цикла. Строится последовательно. Изначально множество *memb* пустое.

15 • Множество *rem_cliques* - набор множеств, которые ещё не использовались для расширения *memb* на предыдущих шагах алгоритма (среди элементов этого набора множеств отсутствуют элементы из *memb*). Изначально *rem_cliques* равен исходному набору множеств.

• Словарь *comp* - словарь, в котором по элементу хранится величина, определяющая, в какое количество множеств входит этот элемент для известного набора элементов (COMP_SET) и заданного набора множеств (*rem_cliques*).

Алгоритм является последовательной процедурой, применяемой к этим трем объектам.

ПРОЦЕДУРА *cover(rem_cliques)*:

25 ПОКА *rem_cliques* НЕ пусто,
ВЫПОЛНЯТЬ:

1 Формируем *comp* по *rem_cliques* и COMP_SET(задаётся извне)

2 Определяем *new_memb*, выбирая из *comp* элемент, входящий в максимальное количество множеств(если таких элементов несколько, выбирается любой из них)

30 3 Добавляем в *memb* значение *new_memb*

4 Формируем *new_rem_cliques*, добавляя в него те множества из *rem_cliques*, в которые не входит *new_memb*

5 Заменяем *rem_cliques* на *new_rem_cliques*

35 *Полнота вычислительного комплекса* - соотношение количества групп, имеющих размер более заданного к общему количеству вычислителей, составляющих вычислительный комплекс.

Временной класс - последовательность временных интервалов заданной ширины, задающая измерительную шкалу, в которой размер интервала является единицей измерения.

40 *Осуществление изобретения*

Для реализации заявляемого изобретения на первом этапе выполнения способа определяют множество вычислителей, которые являются кандидатами в состав вычислительного комплекса. В качестве таких вычислителей могут быть использованы, например, персональные компьютеры, рабочие станции, серверы, мэйнфреймы, кластеры, представляющие собой распределённые системы компьютеров, суперкомпьютеры или облачные ИТ- ресурсы. Каждому вычислителю присваивают идентификационный номер, а также фиксируют его основные характеристики: конфигурацию процессора, объём оперативной и дисковой памяти, операционную

систему, компилятор, файловую систему, сетевой интерфейс.

Затем из определенного известного множества бенчмарков определяют набор бенчмарков, которые планируют запускать на каждом из вычислителей сформированного выше множества. Как правило, набор бенчмарков включает от 10 до 20 тестирующих программ. Рассматриваются популярные бенчмарки, состоящие из программ различных предметных областей <https://www.spec.org/auto/accel/Docs/>. Например, часто используемыми и характерными для своих областей являются бенчмарки, представленные в Таблице 1.

В результате каждый бенчмарк из сформированного набора запускают на каждом вычислителе и фиксируют время его выполнения и соотносят с идентификационным номером вычислителя. На каждом вычислителе из множества вычислителей производят серию, по меньшей мере, из трёх запусков сформированного набора бенчмарков.

Результаты выполнения бенчмарков направляют архитектору вычислительного комплекса, который формирует таблицу результатов запусков бенчмарков на вычислителях. Таблица представляет собой матрицу с усреднёнными результатами запусков, в столбцах которой располагаются идентификационные номера вычислителей, в строках - порядковые номера бенчмарков. Каждая ячейка матрицы с результатами содержит среднее время выполнения бенчмарка на вычислителе (сек.) (см. Таблицу 2, раскрывающую данный этап изобретения на примере конкретного выполнения, где строки 1..10 соответствуют номерам бенчмарков из Таблицы 1, а столбцы 0..24 соответствуют идентификационным номерам вычислителей - кандидатов в вычислительный комплекс).

Таким образом, в результате выполнения данного этапа получают данные о среднем времени выполнения того или иного теста на каждом вычислителе из выбранного множества.

Проводят группировку вычислителей в соответствии со следующим критерием:

Для всего множества вычислителей, являющихся кандидатами в вычислительный комплекс, сформированного на шаге 1), образуют всевозможные пары вычислителей, для чего формируют квадратную матрицу (Таблица 3), строки и столбцы которой представляют собой идентификационные номера кандидатов. Для каждой пары вычислителей определяют коэффициент корреляции Пирсона (к.к.П.) по столбцам с соответствующими временами выполнения бенчмарков (см. Таблицу 3). В случае, если к.к.П. превышает заранее установленное пороговое значение, это означает, что между вычислителями существует линейная зависимость по времени выполнения на них заданного набора бенчмарков. Соответственно, ячейки матрицы содержат значение ккП, определенного для каждой пары вычислителей. КкП может принимать значения от 1 до -1. Отрицательные значения также характеризуют наличие линейной зависимости, поэтому анализируется значение модуля ккП - чем оно ближе к 1, тем больше теснота линейной связи.

Далее процесс ведут для каждого значения ккП, больше 0,1. Задают минимальное пороговое значение ккП в диапазоне от 0,1 до 1 с шагом 0,05, и для каждого такого порогового значения выполняют следующие действия по формированию вычислительного комплекса:

Формируют граф, в качестве вершин которого используют вычислители (их идентификационные номера), при этом рёбрами соединяют пары вычислителей с ккП, превышающим заранее заданное пороговое значение ккП (см. Фиг. 1).

К построенному таким образом графу применяют алгоритм формирования клик (например, алгоритм Брона-Кербоша), в результате применения которого выделяют

полные подграфы, представляющие собой группы вычислителей, попарный модуль значения к.к.П. внутри которых больше заранее заданного порогового значения, например, 0.75. Меняя величину порогового значения ккП при формировании графа вычислителей, можно влиять на количество выделяемых групп вычислителей, и, следовательно, на размер вычислительного комплекса. Таким образом, если величина порогового значения мала, например, 0.2, количество групп в вычислительном комплексе также будет мало, например, 6, что в свою очередь означает незначительное покрытие временных классов программ. Если же порог велик, например, 0.99, то количество групп в вычислительном комплексе также будет велико, например, 17, что в свою очередь означает значительное покрытие временных классов выполняемых программ, однако также может приводить к значительному размеру вычислительного комплекса, что не всегда целесообразно с точки зрения энергозатрат, например. Поэтому для получения оптимального по соотношению число участников/покрытие типов программ вычислительного комплекса необходимо подобрать такое пороговое значение ккП, при котором максимальна величина, равная отношению количества групп более заданного размера к размеру вычислительного комплекса - за счет чего достигается полнота вычислительного комплекса. Однако если выбрать пороговое значение ккП меньше 0.75, то группы могут формироваться случайным образом и не определять временной класс программ. Минимальный размер групп также является значимой величиной, поскольку, например, группа, состоящая из одного узла, может быть аномалией и не определять временной класс.

После группировки вычислителей описанным выше образом, для каждого вычислителя определяют, в какое количество групп он входит. Выбирают вычислитель, входящий в состав максимального количества групп. Затем группы, в которые входит выбранный вычислитель, исключают из рассмотрения, а выбранный вычислитель фиксируют в составе создаваемого вычислительного комплекса. Для оставшихся групп вновь выбирают вычислитель, входящий в состав максимального количества групп. После чего состав групп вновь уменьшается. Процедуру повторяют до тех пор, пока состав групп не исчерпан. В случае если на некотором шаге есть несколько вычислителей, входящих в максимальное количество групп, выбирается любой из них. Выбранные в результате описанной процедуры вычислители объединяют в вычислительный комплекс, который имеет оптимальный состав. Для получения оптимального состава вычислительного комплекса применяют алгоритм формирования состава вычислительного комплекса (например, «жадный» алгоритм).

Для каждого полученного вышеописанным способом вычислительного комплекса проводят оценку его полноты с целью выбора вычислительного комплекса оптимального состава по наибольшему значению полноты, для чего

- определяют минимально возможное количество групп в множестве кандидатов в вычислительный комплекс,
- значение количества групп делят на количество вычислителей сформированного вычислительного комплекса.

Пример конкретного выполнения

В качестве примера конкретной реализации заявляемый способ воплощен в режиме опытной эксплуатации для вычислителей, представленных в [https://www.spec.org/accel/results/accel_omp.html].

Для реализации заявляемого способа сформировано множество кандидатов в состав вычислительного комплекса, состоящее из 25 вычислителей (идентификационные номера 0..24), характеризующихся конфигурацией процессора, объёмом оперативной и дисковой

памяти, операционной системой, компилятором, файловой системой, сетевым интерфейсом (см. например, Таблицу 4 в качестве примера параметров вычислителя).

Для оценки возможности включения указанных вычислителей в вычислительный комплекс сформирован набор, включающий все бенчмарки, описанные ранее. Данный набор считается достаточным, поскольку в нём представлены программы различных предметных областей.

На каждом вычислителе из сформированного множества каждый из выбранных бенчмарков был запущен троекратно, при этом после каждого запуска фиксировали время его выполнения, а также определяли среднее время выполнения каждого теста за три запуска.

Полученные результаты фиксировали на сервере, управляющем формированием вычислительного комплекса. В Таблице 2 представлен пример матрицы результатов выполнения программ на вычислителях, представленных в https://www.spec.org/accel/results/accel_omp.html, и бенчмарков, описанных ранее.

Для полученных результатов измерений определяли коэффициент корреляции Пирсона - попарно, в отношении всех пар вычислителей. В Таблице 3 представлен пример матрицы попарных корреляций. Для дальнейшей группировки вычислителей задавали различные пороговые значения ккП, равного от 0.20 до 0.99 (с шагом 0.01), и определяли состав вычислительного комплекса в зависимости от предустановленного порогового значения.

На фиг. 1 представлен граф вычислителей, в котором рёбрами соединены вычислители, ккП между которыми превышает 0.75 (данное значение выбрано исключительно с иллюстративной целью). В Таблице 5 приведены результаты реализации заявляемого способа для различных пороговых значений ккП. В результате группировки с различными ккП (см. 1 столбец Таблицы 5) выделены группы вычислителей (см. 2 столбец Таблицы 5), попарный ккП которых больше соответствующего заданного порогового значения. Так, для значения ккП 0.2 в графе выделяют 6 групп, а для значения ккП равного 0.99 выделяют 17 групп.

К каждому набору групп применяется алгоритм определения состава вычислительного комплекса, в результате которого определяется состав вычислительного комплекса (см. 3 столбец Таблицы 5).

Таблица 5

Пороговое значение ккП	Группы (идентификационные номера вычислителей)	Состав вычислительного комплекса (ВК)	Полнота ВК
------------------------	--	---------------------------------------	------------

5	1	0.20	[1, 3, 10, 13, 17, 18, 19, 22, 2, 16, 24, 21, 0, 9, 20, 15, 7], [1, 3, 10, 13, 17, 18, 19, 22, 2, 16, 24, 5, 8, 12, 14, 23], [1, 3, 10, 13, 17, 18, 19, 22, 2, 16, 24, 5, 15, 0, 9, 20, 6, 7], [1, 3, 10, 13, 17, 18, 19, 22, 2, 16, 24, 5, 15, 14], [1, 3, 10, 13, 17, 18, 19, 22, 11, 0, 5, 6, 9, 20, 15, 7], [4]	{1, 4}	5/2=2.5
10	2	0.50	[4], [6, 0, 3, 7, 9, 10, 11, 13, 17, 20, 22, 15], [16, 19, 24, 17, 8, 5, 14], [16, 19, 24, 17, 3, 10, 13, 22, 0, 1, 2, 7, 9, 18, 20, 15], [16, 19, 24, 17, 3, 10, 13, 22, 5], [16, 19, 24, 12, 8, 5, 14, 23], [21, 1, 2, 18]	{16, 0, 1, 4}	6/4=1.5
15	3	0.75	[4], [5, 8, 16], [5, 8, 12, 14, 23], [5, 19, 16], [11, 0, 9, 6, 15], [13, 3, 10, 17, 22, 19, 16, 24, 1, 2, 18], [13, 3, 10, 17, 22, 19, 20, 0, 9, 15], [13, 3, 10, 17, 22, 19, 20, 1, 2, 18], [13, 3, 10, 17, 22, 7, 0, 9, 20, 15], [13, 6, 0, 9, 15], [21]	{13, 5, 0, 4, 21}	9/5=1.8
20	4	0.80	[3, 10, 22, 17, 0, 9, 20, 13, 15], [3, 10, 22, 17, 19, 1, 2, 18, 16, 24], [3, 10, 22, 17, 19, 1, 2, 18, 20], [3, 10, 22, 17, 19, 13, 20, 15], [3, 10, 22, 7, 0, 9, 20, 13, 15], [4], [5, 8, 12, 14, 23], [6, 9, 15, 0], [6, 9, 15, 11], [21]	{3, 6, 4, 5, 21}	8/5=1.6
25	5	0.85	[3, 17, 24, 19, 1, 2, 18], [3, 17, 22, 10, 1, 19], [3, 17, 22, 10, 13, 0, 9, 15], [3, 17, 22, 10, 13, 19], [3, 17, 22, 18, 1, 19], [3, 20, 0, 7, 10, 13, 22, 15], [4], [5, 8, 12, 14, 23], [6, 9, 11], [16, 1, 2, 24, 17, 18, 19], [21]	{3, 1, 4, 5, 6, 21}	9/6=1.5
30					
35					
40					
45					

5	6	0.90	[0, 20, 3, 22], [0, 15, 10, 13, 9], [0, 15, 10, 13, 3, 22], [0, 15, 7], [2, 24, 1, 18], [4], [5, 12, 14, 23], [6, 11], [8, 12, 14, 23], [19, 16, 24, 1, 18], [19, 17, 24], [19, 17, 3, 10, 13, 22], [21]	{0, 19, 12, 1, 4, 6, 21}	10/7=1.42
10	7	0.95	[0, 13, 15], [1, 2, 18], [4], [5], [6, 11], [7], [8, 12, 14, 23], [9, 15], [10, 3, 17, 19], [10, 3, 17, 13, 22], [10, 15, 13], [16, 24], [20], [21], [24, 19]	{10, 15, 24, 1, 4, 5, 6, 7, 8, 20, 21}	6/11=0.54
15	8	0.99	[0], [1, 2, 18], [3, 10, 22], [4], [5], [6], [7], [8, 12, 14, 23], [9], [10, 13], [11], [15], [16, 24], [17], [19], [20], [21]]	{10, 0, 1, 4, 5, 6, 7, 8, 9, 11, 15, 16, 17, 19, 20, 21}	3/16=0.18
20					
25					
30					
35					

Была получена оценка полноты каждого сформированного вычислительного комплекса, для чего было поделено значение количества групп, размером более заданного, на количество вычислителей полученного вычислительного комплекса. Наибольшая полнота является характеристикой формирования вычислительного комплекса оптимального состава.

При минимальном размере группы 3 оптимальным порогом ккП среди рассмотренных ккП (от 0.2 до 0.99 с шагом 0.01 - см. фиг. 2) является порог ккП 0.86, поскольку при этом пороге достигается максимальная оценка полноты - 1.83. Такое же значение полноты также достигается при порогах ккП 0.83 и 0.84, однако значение полноты, полученное для более высокого порога 0.86, является более достоверным.

Таким образом, регулируя пороговое значение ккП можно для множества кандидатов в состав вычислительного комплекса сформировать вычислительный комплекс

оптимального состава с максимальной полнотой, то есть вычислительный комплекс, обеспечивающий максимальное покрытие временных классов решаемых задач с помощью минимального количества вычислителей.

СПОСОБ ФОРМИРОВАНИЯ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА

№	Название бенчмарка	Общая категория бенчмарка	Язык программирования	Модель потоков	Дата последнего обновления
1	503.postencil	Термодинамика	C	OpenMP	06 Dec 2016
		https://www.spec.org/auto/accel/Docs/503.postencil.html			
2	504.polbm	Вычислительная гидродинамика, метод решеточных уравнений Больцмана	ANSI C	OpenMP	-
		https://www.spec.org/auto/accel/Docs/504.polbm.html			
3	514.pomriq	Медицина	C	OpenMP	-
		https://www.spec.org/auto/accel/Docs/514.pomriq.html			
4	550.pmd	Молекулярная динамика	Fortran 95	OpenMP	-
		https://www.spec.org/auto/accel/Docs/550.pmd.html			
5	551.ppalm	Моделирование больших вихрей, атмосферная турбулентность	Fortran	-	-
		https://www.spec.org/auto/accel/Docs/551.ppalm.html			
6	552.per	Чрезвычайная параллельность	C	OpenMP	-
		https://www.spec.org/auto/accel/Docs/552.per.html			
7	553.pclvrleaf	Точная гидродинамика	Fortran	OpenMP	24 Oct 2013
		https://www.spec.org/auto/accel/Docs/553.pclvrleaf.html			
8	554.pcg	Сопряженный градиент	C	OpenMP	-
		https://www.spec.org/auto/accel/Docs/554.pcg.html			
9	555.pseismic	Моделирование сейсмических волн	Fortran	OpenMP	-
		https://www.spec.org/auto/accel/Docs/555.pseismic.html			
10	556.psp	Скалярный Пента-Диагональный решатель	Fortran	OpenMP	-
		https://www.spec.org/auto/accel/Docs/556.psp.html			

Таблица 1

СПОСОБ ФОРМИРОВАНИЯ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА

5

10

15

20

25

10	9	8	7	6	5	4	3	2	1	
262.8	251.9	434.8	154.5	806.7	374.5	463.4	378.7	1376.2	197.8	0
54.9	54.2	107.9	60.8	202.5	56.3	149.7	30.5	128.5	25.9	1
53.5	52.6	105.7	50.5	199.6	57.5	139.7	30.2	112.9	25.6	2
83.0	81.5	130.8	61.9	256.7	94.3	230.3	92.8	309.6	52.7	3
2248.6	2360.5	967.0	1325.1	4320.0	434.0	2105.3	1182.6	3382.0	459.7	4
45.6	34.7	48.1	135.3	90.0	76.4	537.2	61.6	259.0	19.0	5
491.6	486.6	760.8	255.6	1314.6	510.0	689.2	1328.6	19732. 3	268.4	6
116.3	109.6	189.7	87.0	338.9	215.5	184.2	214.5	509.3	83.2	7
239.1	185.0	271.0	211.6	578.8	96.2	5223.6	57.1	226.1	157.4	8
80.2	86.4	158.9	73.1	280.0	167.5	199.0	131.0	505.7	60.2	9
84.4	82.7	132.3	73.1	258.8	112.3	269.6	109.1	364.7	53.1	10
44.8	52.2	78.9	49.1	132.8	151.5	223.4	611.6	2147.0	24.2	11
65.6	64.7	101.1	213.5	165.7	64.7	5182.0	57.6	229.0	36.6	12
62.0	56.6	94.1	54.0	170.7	85.2	180.6	83.5	288.0	39.7	13
55.9	42.5	51.2	210.9	102.9	97.4	4654.3	77.5	292.9	22.6	14
126.4	139.3	236.8	101.6	379.6	253.2	342.1	245.6	733.5	83.6	15
46.7	44.8	74.3	56.7	138.5	46.6	191.4	26.2	97.7	30.3	16
60.8	55.3	91.0	53.0	164.8	74.6	194.5	44.7	214.3	31.1	17
54.5	53.9	107.6	57.4	202.1	56.2	152.8	29.7	122.3	25.8	18
53.9	52.7	91.1	53.0	166.3	70.0	209.8	51.0	181.6	40.3	19
136.7	143.4	286.8	101.1	672.7	160.2	282.1	168.9	659.3	169.5	20
875.9	815.7	282.3	329.0	1170.0	230.1	553.6	240.9	621.1	122.6	21
87.0	84.3	135.8	59.0	266.7	100.2	259.8	115.8	359.8	65.8	22
64.7	49.9	216.9	195.5	107.0	58.4	4961.5	58.3	213.5	39.8	23
54.8	52.8	86.7	60.3	163.4	50.9	194.3	32.4	109.2	36.5	24

Таблица 2

СПОСОБ ФОРМИРОВАНИЯ ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА

30

35

40

45

5

10

15

20

13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0.95	0.04	0.78	0.93	0.94	0.06	0.93	0.82	0.35	0.11	0.92	0.70	0.72	1.00	0
0.80	0.36	0.23	0.85	0.66	0.41	0.73	0.28	0.48	0.12	0.90	1.00	1.00	0.72	1
0.77	0.33	0.18	0.83	0.63	0.39	0.73	0.23	0.44	0.12	0.88	1.00	1.00	0.70	2
0.97	0.34	0.58	0.99	0.87	0.37	0.89	0.61	0.56	0.08	1.00	0.88	0.90	0.92	3
0.04	-0.06	-0.02	0.04	0.03	-0.04	-0.00	0.07	-0.08	1.00	0.08	0.12	0.12	0.11	4
0.61	0.91	0.37	0.63	0.48	0.90	0.24	0.32	1.00	-0.08	0.56	0.44	0.48	0.35	5
0.76	-0.05	0.97	0.68	0.86	-0.07	0.63	1.00	0.32	0.07	0.61	0.23	0.28	0.82	6
0.88	-0.02	0.60	0.87	0.80	0.01	1.00	0.63	0.24	-0.00	0.89	0.73	0.73	0.93	7
0.35	1.00	-0.02	0.41	0.15	1.00	0.01	-0.07	0.90	-0.04	0.37	0.39	0.41	0.06	8
0.94	0.14	0.85	0.90	1.00	0.15	0.80	0.86	0.48	0.03	0.87	0.63	0.66	0.94	9
0.99	0.39	0.66	1.00	0.90	0.41	0.87	0.68	0.63	0.04	0.99	0.83	0.85	0.93	10
0.74	0.01	1.00	0.66	0.85	-0.02	0.60	0.97	0.37	-0.02	0.58	0.18	0.23	0.78	11
0.33	1.00	0.01	0.39	0.14	1.00	-0.02	-0.05	0.91	-0.06	0.34	0.33	0.36	0.04	12
1.00	0.33	0.74	0.99	0.94	0.35	0.88	0.76	0.61	0.04	0.97	0.77	0.80	0.95	13
0.34	1.00	0.03	0.40	0.16	0.99	-0.00	-0.03	0.92	-0.09	0.34	0.33	0.35	0.05	14
0.98	0.18	0.81	0.95	0.96	0.19	0.91	0.82	0.49	0.02	0.93	0.69	0.72	0.98	15
0.75	0.72	0.18	0.82	0.56	0.76	0.55	0.20	0.78	0.05	0.83	0.89	0.90	0.56	16
0.96	0.49	0.56	0.98	0.85	0.52	0.79	0.60	0.71	0.09	0.97	0.86	0.89	0.86	17
0.78	0.37	0.20	0.84	0.64	0.43	0.72	0.26	0.49	0.12	0.89	1.00	1.00	0.70	18
0.92	0.59	0.46	0.95	0.78	0.62	0.74	0.48	0.75	0.08	0.95	0.88	0.91	0.79	19
0.87	0.04	0.52	0.89	0.82	0.08	0.89	0.59	0.26	0.14	0.93	0.84	0.84	0.92	20
0.37	0.06	0.08	0.41	0.30	0.11	0.32	0.14	0.13	0.13	0.46	0.58	0.57	0.35	21
0.98	0.35	0.63	1.00	0.89	0.38	0.89	0.65	0.58	0.04	1.00	0.84	0.86	0.93	22
0.33	1.00	0.01	0.38	0.14	0.99	-0.02	-0.05	0.91	-0.08	0.33	0.33	0.35	0.03	23
0.77	0.65	0.18	0.84	0.58	0.69	0.60	0.21	0.71	0.07	0.86	0.93	0.94	0.60	24

Таблица 3

25

30

35

Hardware		Accelerator	
CPU Name:	Intel Xeon E5-2630 v3	Accel Model Name:	Intel Xeon CPU E5-2630 v3
CPU Characteristics:	Intel Turbo Boost Technology up to 3.20 GHz	Accel Vendor:	Intel
	AVX clock: 2100 MHz	Accel Name:	Intel Xeon CPU E5-2630 v3
CPU MHz:	2400	Type of Accel:	CPU
CPU MHz Maximum:	3200	Accel Connection:	N/A
FPU:	Integrated	Does Accel Use ECC:	yes
CPU(s) enabled:	16 cores, 2 chips, 8 cores/chip, 2 threads/core	Accel Description:	Intel Xeon CPU E5-2630 v3 @ 2.40GHz
CPU(s) orderable:	12 chips		Turbo ON, SMT ON
Primary Cache:	32 KB I + 32 KB D on chip per core	Software	
Secondary Cache:	256 KB I+D on chip per core	Operating System:	Ubuntu 14.04.5 LTS
L3 Cache:	20 MB I+D on chip per chip		4.4.0-38-generic
Other Cache:	None	Compiler:	C/C++/Fortran: Version 17.0.2.174 of the Intel Compilers
Memory:	256 GB (16 x 16 GB 2Rx4 PC4-17000R-15, ECC, running at 1866 MHz)	File System:	ext3
Disk Subsystem:	128 GB Samsung SSD 850 PRO	System State:	Run level 5 (user-level)
Other Hardware:	--	Other Software:	

Таблица 4

40

(57) Формула изобретения

1. Способ формирования вычислительного комплекса, включающий следующие этапы:

45

- 1) формирование множества вычислителей, являющихся кандидатами в состав вычислительного комплекса,
- 2) формирование набора бенчмарков, предназначенных к выполнению на каждом вычислителе из множества вычислителей, сформированного на этапе 1),
- 3) серийное выполнение на каждом вычислителе набора бенчмарков с определением и фиксацией времени выполнения каждого бенчмарка из набора,

4) определение среднего времени выполнения в серии каждого бенчмарка отдельно и всего набора бенчмарков в целом на каждом вычислителе из множества вычислителей,

5) составление всех возможных пар вычислителей из множества, сформированного на шаге 1) и определение модуля попарного коэффициента корреляции Пирсона (ккП)

5 для каждой пары на основании данных, полученных на этапах 3) и 4),

и выполнение этапов 6)-9) для каждого из полученных значений модуля (ккП) из диапазона от 0,1 до 1 с шагом от 0,01 до 0,05:

6) выделение групп вычислителей, включающих пары вычислителей, имеющих значение модуля попарного коэффициента корреляции Пирсона (ккП) больше заданного значения,

7) определение множества вычислителей из вычислителей, вошедших в группы, сформированные на шаге 6), включающего по одному вычислителю из всех групп, выделенных на шаге 6),

8) формирование вычислительного комплекса из вычислителей, определенных на шаге 7), с исключением повторных включений одного и того же вычислителя в комплекс,

9) определение полноты сформированного вычислительного комплекса, для чего для каждого сформированного на шаге 8) вычислительного комплекса определяют отношение количества групп, полученных на шаге 6), к количеству вычислителей, составляющих множество на шаге 1),

20 после чего выбирают наиболее оптимальный вычислительный комплекс по наибольшему значению его полноты, определенной на этапе 9).

2. Способ по п.1, отличающийся тем, что при формировании набора бенчмарков для выполнения этапа 2) выбирают бенчмарки в количестве не менее 5 штук: а)

25 принадлежащие различным временным классам, то есть бенчмарки, время выполнения которых на одном и том же вычислителе находится в разных измерительных временных шкалах, б) характеризующиеся минимальным и максимальным временем выполнения, в) характеризующиеся различными значениями требуемых машинных ресурсов для выполнения.

3. Способ по п.1, отличающийся тем, что серийное выполнение набора бенчмарков на каждом вычислителе включает, по меньшей мере, три последовательных запуска набора.

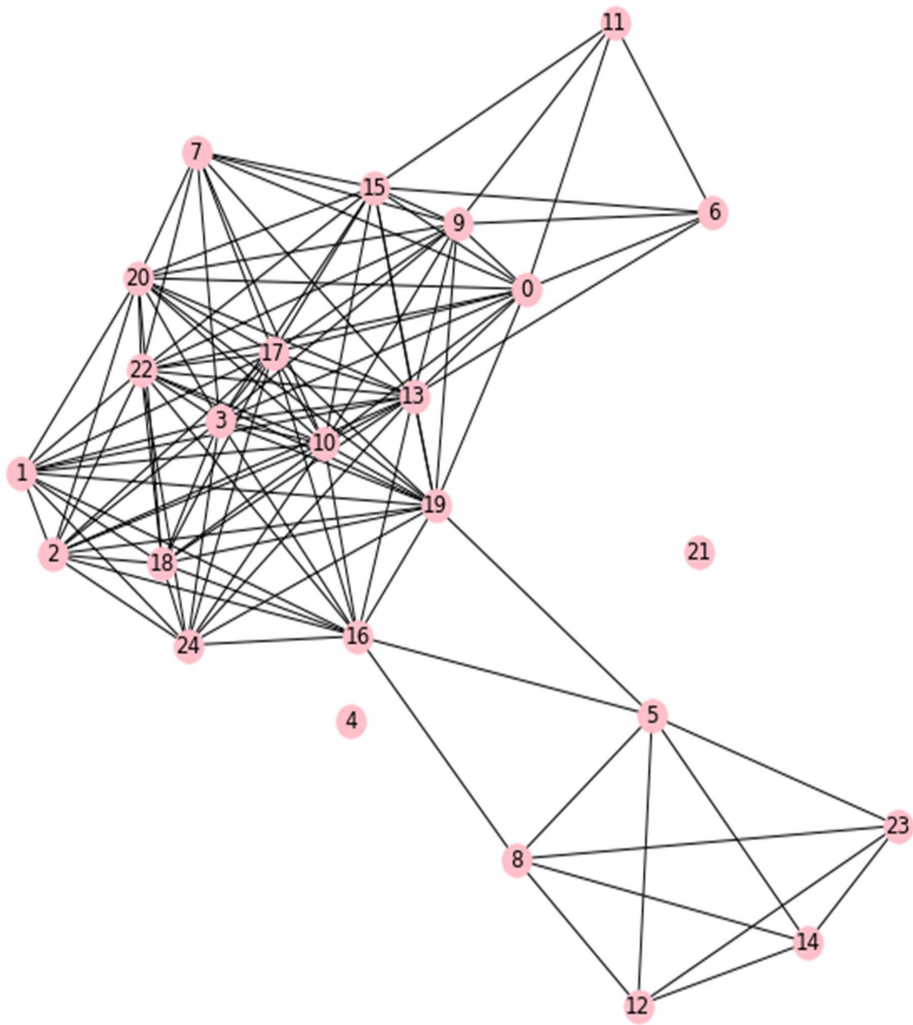
4. Способ по п.1, отличающийся тем, что на этапе 6) для формирования групп вычислителей используют алгоритм формирования клик.

35 5. Способ по п.4, отличающийся тем, что при этом в качестве алгоритма формирования клик используют алгоритм Брона-Кербоша.

40

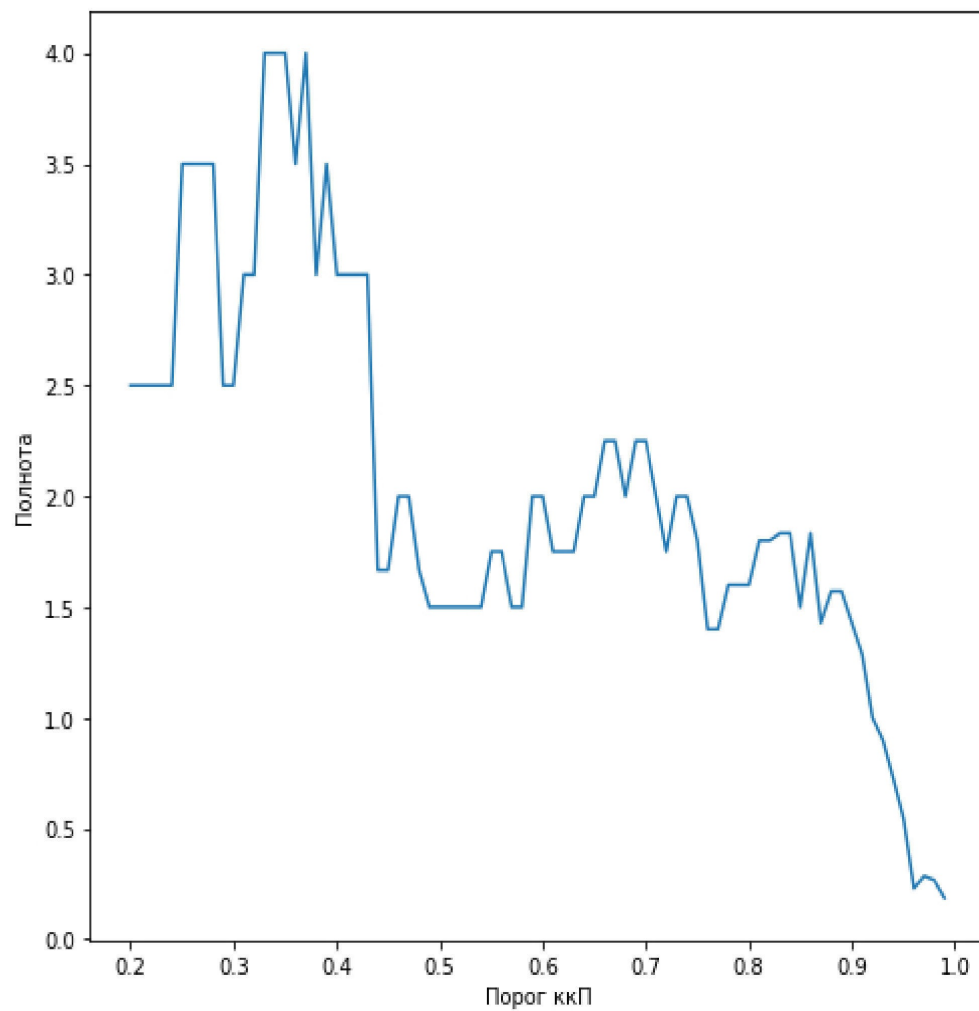
45

1



Фиг. 1

2



Фиг.2