

8. *Хайкин С.* Нейронные сети: полный курс. – М.: Издательский дом «Вильямс», 2006. – 55 с.
9. Итоги науки и техники: физические и математические модели нейронных сетей, том 1. – М.: Изд-во ВИНТИ, 1990.
10. *Коновалов О.В.* Разработка и исследование генетических методов обучения нейронных сетей для задач визуализации в САПР-К / Диссертация на соискание ученой степени канд. техн. наук. – Таганрог, 2003. – 136 с.

Чижков Александр Васильевич

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: 4ijkov@rambler.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 8(8634)371-651.

Кафедра систем автоматизированного проектирования.

Аспирант.

Сейтова Светлана Владимировна

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 8(8634)371-606.

Кафедра высшей математики.

Аспирант.

Chizhkov Aleksandr Vasilievich

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: 4ijkov@rambler.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 8(8634)371-651.

Department of Computer Aided Design.

Post-graduate student.

Seitova Svetlana Vladimirovna

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 8(8634)371-606.

Department of Higher Mathematics.

Post-graduate student.

УДК 681.518

В.А. Литвиненко, С.А. Ховансков, О.Р. Норкин

**ОПТИМИЗАЦИЯ МУЛЬТИАГЕНТНОЙ СИСТЕМЫ РАСПРЕДЕЛЕННЫХ
ВЫЧИСЛЕНИЙ***

Современные темпы развития средств вычислительной техники не удовлетворяют требованиям предъявляемых ростом сложности решаемых в настоящее время задач. Для решения задач исследовательского характера, моделирования,

* Работа выполнена при поддержке: РФФИ (грант № 09-07-00318), г/б № 2.1.2.1652.

выбора оптимального решения большой объем вычислений не позволяет получить нужный результат за приемлемое время. Задачи, для которых временной критерий является наиболее важным, усложняются и их объем вычислений продолжает увеличиваться. Все это происходит более быстрыми темпами, чем рост производительности вычислительных средств и создает проблему соблюдения временных критериев задачи.

Известны следующие направления преодоления проблемы уменьшения времени решения задачи: первое – снижение времени выполнения операций за счет развития аппаратных средств вычислительной техники; второе – создание распределенных вычислительных систем или систем распределенных вычислений [1].

Первое направление уже не позволяет удовлетворить требуемым критериям производительности при решении большеобъемных задач.

Второе направление представляется более перспективным. Здесь существует несколько направлений:

- ◆ использование многоядерных процессоров;
- ◆ многопроцессорные вычислительные машины;
- ◆ многомашинные системы (кластеры).

На этом направлении существуют свои проблемы, которые затрудняют использование распределенных вычислительных систем. Здесь основной проблемой является не создание специальных аппаратных средств (хотя она присутствует) а программная организация на аппаратных средствах распределенных вычислений [2].

Организация заключается в разделении всего объема вычислений на программные модули, распределении этих модулей между центрами обработки данных, синхронизация их работы и порядок реализации процессов обмена данными между центрами.

Особенности организации связаны со степенью связности данных, со способом реализации вычислений, а также с управлением процессом выполнения задачи на всех центрах обработки данных и их синхронизации.

Например, на многопроцессорной системе выполнение вычислительных модулей организовывается специальной программой, распределяющей между процессорами нагрузку, обеспечивающей их взаимодействие и слежение за работоспособностью и ходом выполнения вычислений.

В многомашинных системах также должен быть центр, координирующий работу всех машин.

С помощью современных специальных языков программирования можно отказать от центра управления, а выполнение задачи организовать на нескольких машинах путем одновременного выполнения всех модулей задачи.

Помимо общей для всех методов проблемы разделения задачи на модули существуют проблемы стоимости (спецвычислители), наличие специальных центров управления, жесткое назначение каждому центру вычислительного модуля задачи, невозможность быстрой перестройки системы на выполнения другой задачи.

Поиск решение этих проблем привел к появлению еще одного пути реализации распределенных вычислений – использование мультиагентной системы, реализующей принцип коллективного принятия решения. Агентом является программный код, выполняемый в фоновом режиме на локальном компьютере. Каждый компьютер, имеющий одного из агентов, работает независимо от других аналогичных компьютеров. При выполнении распределенных вычислений агенты мультиагентной системы способны обмениваться между собой данными.

Мультиагентная система экономичнее всего реализуется на наиболее доступной вычислительной системе – компьютерной сети, в которой при организации распределенных вычислений каждому вычислительному блоку ориентированного графа задачи ставится в соответствие компьютер с учетом наличия физических связей в сети, способных реализовать потоки передаваемой информации между вычислительными блоками.

Основной алгоритм программы-агента реализует следующие задачи:

- ◆ получение информации от компьютеров-источников;
- ◆ выполнение на компьютере требуемых операций над полученной информацией;
- ◆ передача результата компьютеру-получателю.

Структура ориентированного графа решаемой задачи представляет собой совокупность подмножеств связанных вершин. Эти подмножества образуют иерархическую структуру с несколькими уровнями.

Самым верхним уровнем является вершина графа, получающая выходные данные. Подмножества образуют узлы нижнего уровня и узлы высокого уровня, которым передаются данные.

Другие подмножества содержат вершину верхнего уровня вместе с связанными с ней вершинами более нижнего уровня.

Самым нижним уровнем является вершина, принимающая входные данные.

Вся информация передается между агентами порциями. Обработку одной порции данных можно считать одним шагом процесса обработки всей системой. Агенту, передавшему обработанные данные, поступает новая порция данных, и выполняется следующий шаг обработки. Передача данных в сети от входного компьютера к выходному выполняется в пульсирующем режиме.

Обмен данными между агентами включает в себя не только передачу данных задачи, но и служебную информацию [3,4].

После пересылки обработанных данных агенту более высокого уровня выполняется обмен между агентами подготовленной заранее информацией. Обмен информацией производится между агентами, расположенными только на компьютерах одного либо соседних подмножеств.

Служебная информация используется для минимизации времени обработки пакета данных на одном шаге. Время обработки данных в компьютере зависит от цифровых параметров компьютера и свойств программного обеспечения, а также от характера обработки данных [4].

В системе временем начала обработки данных на компьютерах агентов текущего уровня будет считаться время приема последнего требуемого пакета данных от агентов нижнего уровня связанных с ними. Соответственно время, за которое на этом пути появятся обработанные на предыдущем уровне данные, и будет временем выполнения предыдущего шага.

Оценка времени выполнения вычислений на одном компьютером производится на основании работы с соответствующей порцией информации, поэтому эта оценка является индивидуальной и может использоваться для выполнения оптимизации системы [3].

На ориентированном графе задачи существует несколько маршрутов передачи обработанных данных от входного компьютера к выходному. Из них можно выделить тот путь передачи данных от агента самого нижнего уровня к агенту са-

мого высокого уровня, который и определяет время обработки данных всей вычислительной системой. Такой путь называется критическим путем.

$$T_{кр} = \max(T_1, T_2, \dots, T_i, \dots, T_k),$$

где k - количество путей между входным агентом $a_{вх}$ и выходным агентом $a_{вых}$.

В предполагаемом алгоритме оптимизируется время выполнения системой задачи путем подбора назначения графа задачи на граф сети.

Алгоритм агента.

1. Прием информации от агентов уровня $n - 1$.
2. Прием служебной информации от агентов уровня $n + 1$.
3. Определение критического пути.
4. Подбор варианта обмена для агента $k_i^{n+1,c}$.
5. Все варианты обмена использованы? Если «Да», то перейти к п. 7, иначе – к п. 6.
6. Передача служебной информации агентам уровня $n - 1$.
7. Обработка данных..
8. Обработка данных закончена? Если «Да», то перейти к п. 9, иначе – к п. 7
9. Передача всей информации агентам уровня $i + 1$.
10. Переход к п. 1.

Все множество вершин ориентированного графа задачи разбивается на непесекающиеся подграфы путем их раскраски G . Для раскраски графа используются функциональные признаки вычислительных блоков.

Сначала оптимизация выполняется одновременно внутри подграфов между циклами обработки данных. Затем выполняется оптимизация между соседними подграфами.

При этом время, затрачиваемое агентами подграфа на оптимизацию, не зависит от мощности подграфа, что позволяет отвести под процесс оптимизации во всех подграфах фиксированный промежуток времени.

В следующий цикл обмена информацией между агентами соседних подграфов выполняется аналогичная оптимизация в соседних подграфах.

Процесс оптимизации между агентами в одном подграфе многократно чередуется с оптимизацией между агентами в соседних подграфах. Этот процесс будет постоянно повторяться, поскольку процесс будет запускаться при каждой передаче данных между агентами вычислительных блоков во время выполнения задачи, то он будет вестись постоянно. Время, затрачиваемое на оптимизацию, будет сведено к минимуму.

Передача данных производится от нижнего уровня на более верхние согласно графу задачи G . Самым верхним уровнем является выходной компьютер. Все множество P компьютеров с агентами, реализующими граф задачи, разбиты на подмножества вершин по уровням $P = \{P_1, P_2, \dots, P_n\}$, которые, в свою очередь, могут состоять из более мелких подмножеств $P_n = \{p^{n,1}, p^{n,2}, \dots, p^{n,b}\}$. Любое подмножество вершин $p^{n,b}$ состоит из агентов на компьютерах $(k_1^{n,b}, k_2^{n,b}, \dots, k_m^{n,b})$, соединенных по графу задачи с одним из компьютеров верхнего уровня $k_i^{n+1,c}$.

Подмножеству компьютеров $(k_1^{n,b}, k_2^{n,b}, k_3^{n,b}, \dots, k_m^{n,b})$ перед выполнением первого шага задачи назначено подмножество $(g_1^{n,b}, g_2^{n,b}, g_3^{n,b}, \dots, g_m^{n,b})$ графа задачи G .

После выполнении первого шага задачи на компьютерах подмножества $p^{n,b}$ выполняется передача результатов вычислений вместе с управляющей информацией агенту на компьютере $k_i^{n+1,c}$. Собирается информация от каждого агента о времени выполнения этого шага

$$(t_1^{n,b} \in k_1^{n,b}, t_2^{n,b} \in k_2^{n,b}, t_3^{n,b} \in k_3^{n,b}, \dots, t_m^{n,b} \in k_m^{n,b})$$

и выбирается компьютер $k_y^{n,b}$ имеющий $t_{\min 1}^{n,b} = \min(t_1^{n,b}, t_2^{n,b}, \dots, t_m^{n,b})$. После поступления управляющей информации от агента компьютера $k_i^{n+1,c}$ агенты подмножества $p^{n,b}$ получают информацию, принятую от всех компьютеров $p^{n,b}$. Выполняется второй шаг решения задачи.

После выполнении второго шага задачи на компьютерах подмножества $p^{n,b}$ выполняется передача информации агенту на компьютере $k_i^{n+1,c}$, как и на первом шаге. Также собирается информация о времени выполнения этого шага всеми компьютерами и выбирается компьютер $k_x^{n,b}$, имеющий

$$t_{\min 2}^{n,b} = \min(t_1^{n,b}, t_2^{n,b}, \dots, t_m^{n,b}).$$

Компьютер $k_i^{n+1,c}$ оценивает время завершения первого шага решения и второго. Если $t_{\min 1}^{n,b} > t_{\min 2}^{n,b}$, то принимается решение о выполнении нового переназначение агентам подмножества $p^{n,b}$, аналогичное выполненному перед вторым шагом.

Одновременно, точно по такому алгоритму, выполняются переназначения и во всех подмножествах всех уровней.

Вся информация собирается у агента $a_{блх}$. Этой информации достаточно, чтобы определить критический путь и выделить в нем подмножество компьютеров, определяющее время выполнения шага системой. Агент $a_{блх}$ выбирает подмножество, имеющее минимальное время выполнения шага и передает указание подобрать компьютеры для обмена между подмножествами так, чтобы максимальное критическое время системы сократилось.

На рис. 1-4 приведен пример работы алгоритма на небольшом отрезке времени выполнения обработки данных системой, состоящей из 10 компьютеров, отображены параметры узлов графов задачи и сети. Рис. 1,2 и рис. 3,4 показывают состояние узлов системы до начала выполнения алгоритма и после соответственно.

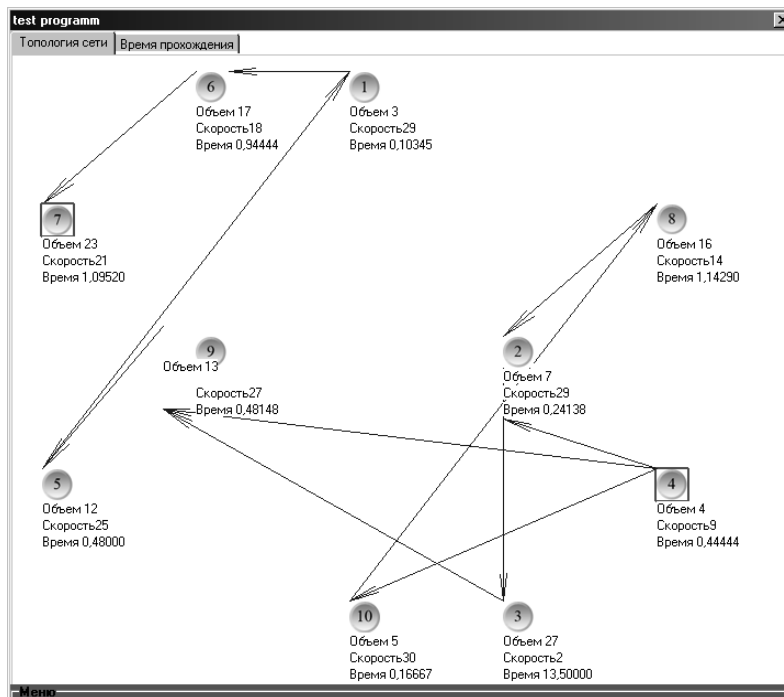


Рис. 1. Начальное размещение узлов системы

The screenshot shows the 'test programm' window with the 'Время прохождения' tab selected. It displays three paths of information flow with their respective time parameters.

Путь №	Узлы	Время	Общее время
Путь №1	1 10 4 5 6 3 7 8 2 9	0.10 0.16 0.44 0.48 0.94 13.5 1.09 1.14 0.24 0.48	18.60000
Путь №2	1 0 0 5 6 3 7 8 2 9	0.10 0 0 0.48 0.94 13.5 1.09 1.14 0.24 0.48	17.98900
Путь №3	1 0 0 0 0 3 7 8 2 9	0.10 0 0 0 0 13.5 1.09 1.14 0.24 0.48	16.56400

Рис. 2. Начальные временные параметры узлов системы

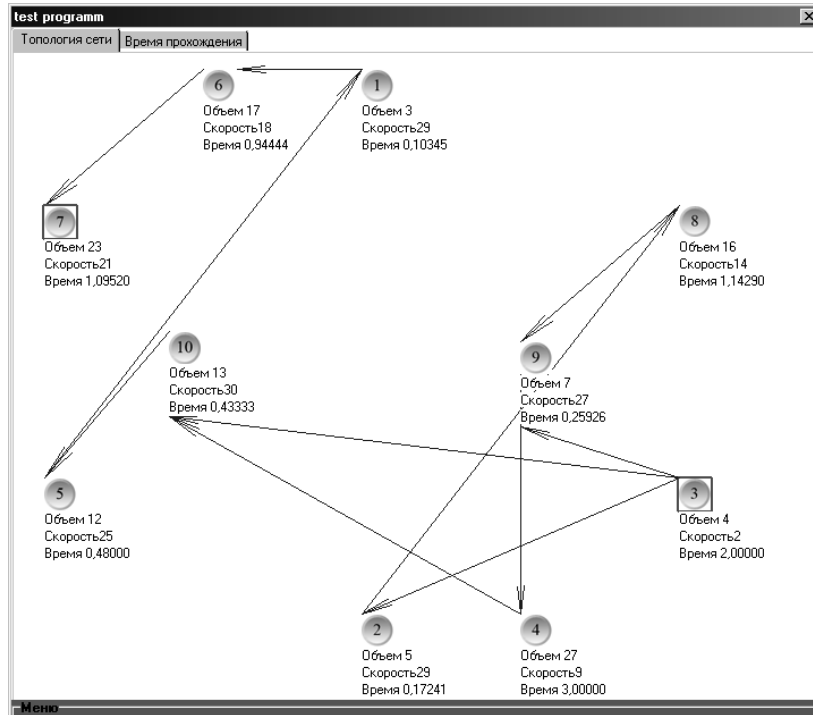


Рис. 3. Конечное размещение узлов системы

test programm										
Пути прохождения информации										
Узлы										
Путь №1	1	2	3	5	6	4	7	8	9	10
Время	0,62	0,57	0,55	0,70	2,00	2,42	1,26	0,57	0,70	1,40
Общее время прохождения по первому пути										10,81700
Путь №2	1	0	0	5	6	4	7	8	9	10
Время	0,62	0	0	0,70	2,00	2,42	1,26	0,57	0,70	1,40
Общее время прохождения по второму пути										9,68960
Путь №3	1	0	0	0	0	4	7	8	9	10
Время	0,62	0	0	0	0	2,42	1,26	0,57	0,70	1,40
Общее время прохождения по третьему пути										6,98960

Рис. 4. Конечные временные параметры узлов системы

Алгоритма назначения вершин графа задачи G компьютерам сети под управлением мультиагентной системой позволяет выполнять оптимизацию при любом первоначальном назначении графа G на сеть. Этот алгоритм не требует предварительной оптимизации времени решения компьютерами сети. Оптимизация выполняется уже во время решения самой задачи, используя для этого реальные данные. Отсутствие центра управления системой и способность к адаптации при изменении состава сети позволяет обеспечить системе высокую степень живучести.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Таненбаум Э.* Распределенные системы: принципы и парадигмы. – СПб: Питер, 2003. – 877 с.
2. *Ховансков С.А., Литвиненко В.А., Калашников В.А.* Использование многопроцессорной вычислительной системы для решения задачи трассировки // Ведомственный тематический сборник "Интеллектуальные САПР". – Таганрог: Изд-во ТРТУ, 1994, Вып.4.
3. *Ховансков С.А., Литвиненко В.А.* Организация распределенных вычислений на основе мультиагентного подхода // Известия ТРТУ. Тематический выпуск «Интеллектуальные САПР». – Таганрог: Изд-во ТРТУ, 2007. №1(73). – С. 246-250
4. *Ховансков С.А., Литвиненко В.А.* Решение задач путем организации распределенных вычислений в сети // Известия ЮФУ. Технические науки. – Таганрог: Изд-во ТТИ ЮФУ, 2008, №3(80). – С. 16-21.

Литвиненко Василий Афанасьевич

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: litv@tsure.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 8(8634)371-651.

Кафедра систем автоматизированного проектирования; доцент.

Ховансков Сергей Андреевич

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: litv@tsure.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 8-918-57-92-173.

Кафедра радиоэлектронных средств защиты и сервиса; доцент.

Норкин Олег Рауфатович

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: oran@tsure.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 8(863)371-787.

Кафедра системного анализа и телекоммуникаций; доцент.

Litvinenko Vasilij Afanasievich

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: litv@tsure.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 8(8634)371-651.

Department of Computer Aided Design; associate professor.

Hovanskov Sergey Andreevich

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: litv@tsure.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 8-918-57-92-173.

Department of Electronic Means of Protection, Security and Services; associate professor.

Norkin Oleg Raufatovich

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: oran@tsure.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 8(863)371-787.

Department of System Analysis and Telecommunications; associate professor.