

ОРГАНИЗАЦИЯ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ В СИСТЕМЕ ПРОГРАММИРОВАНИЯ ПАРСЕК НА ПРИМЕРЕ СЖАТИЯ ЦИФРОВОГО ВИДЕО¹

Ю.С. Затуливетер, Е.А. Фищенко

Институт проблем управления им. В.А. Трапезникова, г. Москва

Рассмотрен новый подход к организации распределенных вычислений в сети Интернет, основанный на математически полной и замкнутой модели исчисления древовидных структур, реализованной посредством системы программирования ПАРСЕК. В язык ПАРСЕК введены операторы управления ресурсами и групповой передачи данных. На примере кодирования сжатия в реальном времени для видеоконференций в стандарте H.261 и цифрового видео в стандарте MPEG показана эффективность предложенного способа управления сетевыми ресурсами.

ВВЕДЕНИЕ

Следующий шаг в развитии Интернета (далее Сети) – предоставление пользователю суперкомпьютерных возможностей, что позволит ему решать задачи, требующие $10^{10} \dots 10^{18}$ операций в секунду и более, за счет совместного использования средств обработки информации одновременно на многих компьютерах Сети. В связи с этим стала весьма актуальной задача разработки средств регулярного программирования глобально распределенных вычислений.

Движение к глобальному программированию в Сети возможно по двум встречным направлениям. Одно – «снизу – вверх» (от сложившейся практики), другое – «сверху – вниз» (от новой теории).

Пример первого направления – Grid-технологии [1], подход к виртуальной организации вычислительных ресурсов Сети для распределенных вычислений путем создания глобально распределенной операционной системы. Это путь постепенного наращивания и расширения возможностей существующих средств индустриального программирования в рамках устоявшихся компьютерных платформ. Многочисленные и разнородные версии Grid-технологий толерантны к классической компьютерной парадигме, потому органично вписываются в арсенал индустриального программирования. Интеграция глобально распределенных вы-

числительных ресурсов осуществляется за счет внешних программных надстроек посредством силового преодоления комбинаторного сопротивления информационного шума, порождаемого разнородной компьютерной информацией.

Встречное направление начинается с новой концепции математически однородного поля компьютерной информации, в которой на уровне аксиоматики устранены причины воспроизводства информационного шума [2]. В этой концепции проблемы программирования, интеграции и масштабирования компьютерных решений и распределенных процессов перестают зависеть от технических особенностей компьютеров и сетей. Регулярный инструмент глобального программирования строится сначала на теоретическом уровне в виде математически полной и замкнутой модели исчисления древовидных структур, а затем алгоритмы функционирования модели программируются в ресурсах Сети посредством языка ПАРСЕК [3], воплощающего исчисление древовидных структур.

ОСОБЕННОСТИ СИСТЕМЫ ПРОГРАММИРОВАНИЯ ПАРСЕК

Различные формы древовидных структур широко применяются в языках программирования как формализованные типы данных, например, списки (Лисп). Их можно представить через общие структуры (Си, Паскаль) или через объекты (С++, Java). В прикладных системах с помощью древовидных структур реализуются удобные пользовательские интерфейсы, гипертекстовые спра-

¹ Работа выполнена при поддержке РФФИ, проект № 01-01-01047



вочки и др. Практика показывает: деревья упрощают пользование программными продуктами.

Язык и система программирования ПАРСЕК реализует исчисление древовидных структур с открытой интерпретацией, представленных для программиста в форме вертикальной ленты строк («одна строка – одна вершина»), которую еще Питер Нортон ввел в массовый компьютерный обиход, дав геометрическое представление файловой системы. В языке ПАРСЕК деревья в подобной геометрической форме дают универсальное и математически замкнутое представление компьютерной информации – как данных, так и программ. Решая задачи, программист создает необходимые деревья и функции их преобразования, по своему усмотрению назначая смысловую нагрузку и деревьям, и функциям их преобразования. Полнота и замкнутость исчисления древовидных структур в сочетании с их открытой содержательной интерпретируемостью дают практическую универсальность в применении к различным задачам. В значительной мере упрощается параметризация программ. На идеологическом уровне исключена опасность разрушения информации в памяти из-за переполнения индексных полей. При полностью динамическом управлении памятью ее расход ограничен заранее известными и приемлемыми для практики оценками.

СПОСОБ УПРАВЛЕНИЯ СЕТЕВЫМИ РЕСУРСАМИ

Для управления сетевыми ресурсами при организации распределенных вычислений создается специализированная база данных доступных ресурсов, с помощью которой производится динамическое выделение и освобождение ресурсов. В ней содержится информация о компьютерах с указанием их производительности и объемов памяти, пропускной способности маршрутов в сети и т. д.

От задачи идут запросы на выделение ресурсов: число компьютеров, размеры памяти, объемы и темп передачи данных и др. Формируется расписание использования выделенных компьютеров (рис. 1), в которых выполняются вычисления. По мере освобождения выделенных ресурсов они возвращаются базе данных для повторного использования.

Компьютер, посылающий базе данных запросы на выделение ресурсов, называется ведущим (ведущий 1 – см. рис. 1). Он также осуществляет пуск задачи, динамическое распределение заданий, ввод/вывод данных, сбор результатов и освобождение ресурсов. Распределенные задания реализуют компьютеры сети, называемые ведомыми (ведомые 1.1, 1.2, 1.3, 1.4), которые могут быть ведущими (ведущие 2.1, 2.2) в отношении подзадач (ведомые 2.1.1, 2.1.2, 2.2.1, 2.2.2).

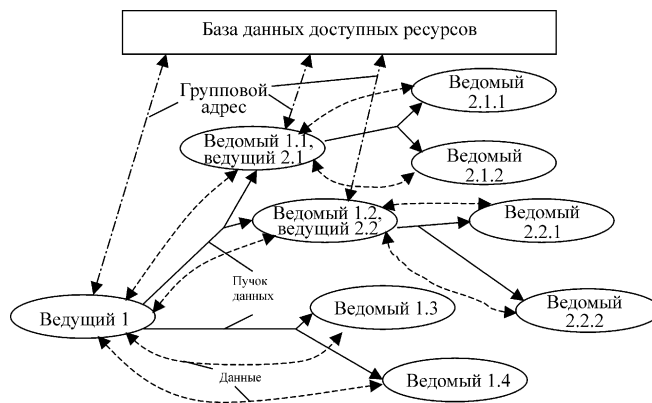


Рис. 1. Дерево заданий

В данной работе для эффективного управления распределенными вычислениями в сети предлагается ввести возможность работы группы ведомых компьютеров по одной программе и использовать функции передачи данных по групповому адресу, что позволит уменьшить время передачи данных и ускорить общее время решения задачи.

Цель данной работы – исследовать возможности предложенного способа управления сетевыми ресурсами для хорошо распараллеливаемых задач, требующих производительности вычислительных ресурсов $10^{10} \dots 10^{14}$ оп/с. В качестве примеров рассмотрим задачи сжатия цифрового видео в реальном времени для видеоконференций в стандарте H.261 и кодирования в стандарте MPEG.

На примере задач высокой вычислительной сложности реального времени предлагается схема реализации распределенного счета применительно к системе ПАРСЕК, библиотека которой в поддержку распределенных вычислений расширяется следующими функциями работы с сетью:

$gip = G_alloc(n)$ – выделение группы из n свободных компьютеров для исполнения n заданий. Компьютер посылает в базу данных запрос на выделение группы IP-адресов свободных компьютеров. Ответ на запрос gip – либо список адресов, либо групповой адрес, представляющий собой сетевой адрес с маскированными разрядами;

$G_free(gip)$ – освобождение занятых компьютеров по групповому адресу gip ;

$Send(name1, name2, \dots, nameM, gip)$ – пересылка данных с именами $name1, name2, \dots, nameM$ по групповому адресу gip ;

$Wait_D(name1, name2, \dots, nameM)$ – оператор ожидания доставки данных с именами $name1, name2, \dots, nameM$, отсылаемых функциями $Send(\cdot)$; этот оператор необходим для получения полного набора входных данных для запуска заданий на исполнение.

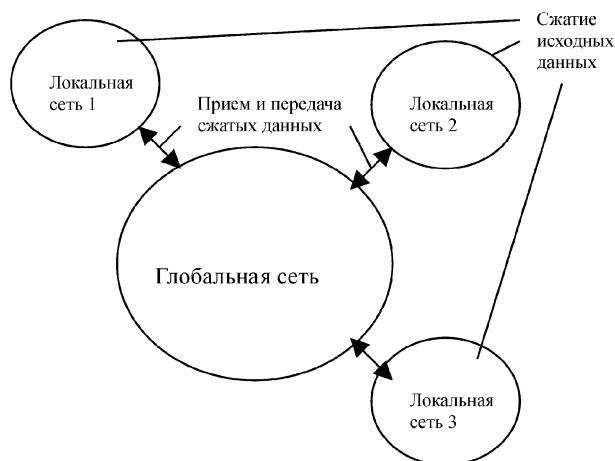


Рис. 2. Организация распределенных вычислений для видеоконференций

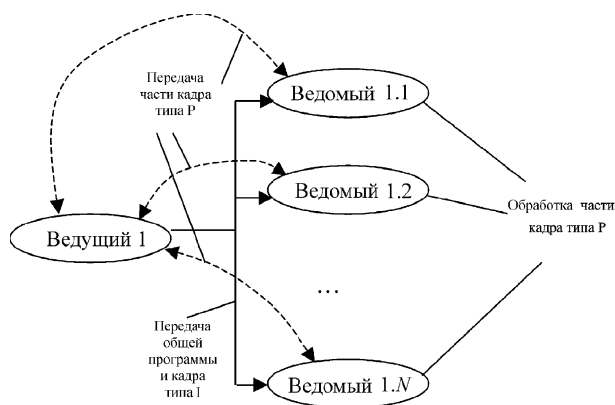


Рис. 3. Организация распределенных вычислений в локальной сети

Процедурный язык ПАРСЕК создавался под задачи с динамически сложными структурами, поэтому, расширяя его возможности, мы распространяем достоинства этого языка на решение задач в глобально распределенных ресурсах.

РАСПРЕДЕЛЕННЫЕ ВЫЧИСЛЕНИЯ ДЛЯ ВИДЕОКОНФЕРЕНЦИЙ В СТАНДАРТЕ H.261

Для видеоконференций в хорошо известном стандарте H.261 [4] построим распределенный алгоритм сжатия цифрового видео. Для получения качественного видеоряда видеоконференций для сжатия изображений в реальном времени требуется скорость обработки более 10^{10} коротких операций в секунду.

Требуемая скорость обработки рассчитана, исходя из выполнения наиболее трудоемкой операции поиска векторов движения для кадров формата CIF. Кадры состоят из макроблоков размером $m \times m$, для каждого из них необходимо осуществить

поиск векторов движения в окне. Число операций O_s , которые необходимо выполнить за одну секунду, определяется следующим образом: $O_s = 4FACW^2o_{mb}$, где F – число кадров, поступающих в секунду, A и C – число пикселей по горизонтали и вертикали в кадре, W – размер окна поиска в макроблоках, o_{mb} – число восьмиразрядных операций для сравнения пары макроблоков. При $F = 30$, $A = 352$, $C = 288$, $W = 1$, $o_{mb} = 768$ число операций в секунду равно 10^{10} . Если на выполнение одной восьмиразрядной операции тратится $0,25 \cdot 10^{-9}$ с, то на обработку данных, поступающих за одну секунду, потребуется 2,5 с, что не удовлетворяет условиям реального времени.

Для выполнения этих условий рассматриваются возможности реализации сжимающего кодирования видеоконференций параллельной обработкой на нескольких компьютерах сети. Для этого на стороне каждого из участников видеоконференции на компьютерах своей локальной сети производится распределенная обработка аудио- и видеoinформации. Сжатый в реальном времени (путем параллельной обработки) поток информации посылается через глобальную сеть другим участникам видеоконференции, которые обрабатывают потоки данных в компьютерах своих локальных сетей (рис. 2).

Кодирование в локальной сети производится следующим образом (рис. 3). Каждый из участников видеоконференции должен иметь сеть с числом компьютеров $N + 1$ (значение N будет рассчитано далее), обменивающихся между собой со скоростью передачи данных не менее 10 Мб/с. Оцифрованные кадры поступают в ведущий компьютер, где производится их внутреннее кодирование в соответствии со стандартом MJPEG [4].

Время внутреннего кодирования T_i для кадров, поступающих за одну секунду, определяется как $T_i = F \frac{A}{8} \frac{C}{8} o_b t_i$, где o_b – число операций для одного блока, t_i – длительность одной операции. При $o_b = 1000$, $t_i = 10^{-9}$ с время внутреннего кодирования займет 0,05 с. Внутреннее кодирование позволяет в 8 раз сократить объемы данных, передаваемых ведомым компьютерам, и снизить требования к пропускной способности локальной сети.

Затем первый кадр группы, который должен быть преобразован в кадр типа I, групповым образом (один ко многим) передается N ведомым компьютерам. Последующие кадры типа P, для которых должно быть выполнено межкадровое кодирование, разбиваются на N частей. Каждая часть, содержащая $Y = \frac{AC}{m^2 N}$ макроблоков, передается ответственному ведомому компьютеру.

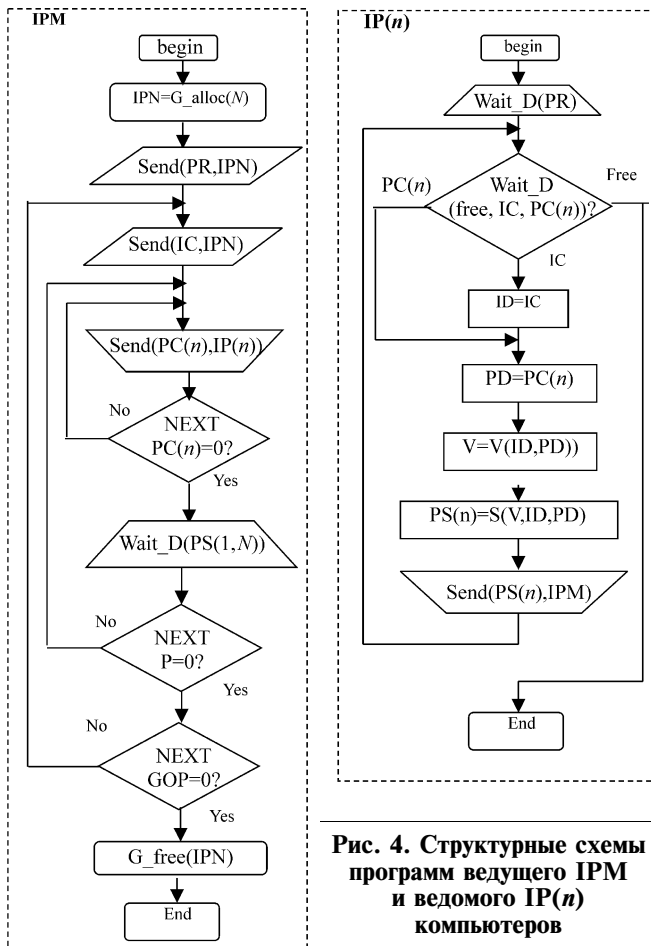


Рис. 4. Структурные схемы программ ведущего IPM и ведомого IP(n) компьютеров

Время S_i передачи кадров с внутренним кодированием, поступающих за одну секунду, определяется следующим образом: $S_i = \frac{FACp}{vk_i}$, где p – число разрядов в пикселе, k_i – коэффициент сжатия при внутреннем кодировании, v – скорость передачи данных в локальной сети. Для $p = 8$, $k_i = 8$, $v = 10$ Мб/с время передачи кадров с внутренним кодированием составляет 0,3 с.

Сжатые с помощью межкадрового кодирования кадры типа P затем возвращаются ведущему компьютеру. Время S_{ni} передачи сжатых с помощью межкадрового кодирования кадров определяется следующим образом: $S_{ni} = \frac{FACp}{vk_i k_{ni}}$, где k_{ni} – коэффициент сжатия при межкадровом кодировании. Для $k_{ni} = 3$ время передачи кадров с межкадровым кодированием S_{ni} , поступающих за одну секунду, составляет 0,1 с.

В каждом ведомом компьютере производится декодирование кадра типа I, декодирование и межкадровое кодирование Y макроблоков кадра

типа P, время счета $T_{ni} = 4FYW^2 o_{mb} t_{mb}$. Для $t_{mb} = 0,25 \cdot 10^{-9}$ время счета $T_{ni} = 0,005 \cdot Y$.

Общее время обработки T_s для F кадров, поступающих за одну секунду, определяется следующим образом: $T_s = T_i + T_{ni} + S_i + S_{ni}$. Для выполнения условий реального времени $T_s \leq 1$; $T_s = 0,005Y + 0,1 + 0,3 + 0,05 \leq 1$, $Y \leq 110$, $N \geq 4$.

В данном случае число ведомых компьютеров должно быть не менее четырех.

Структурные схемы программ для ведущего компьютера с именем IPM и ведомого компьютера с именем IP(n) приведены на рис. 4. Принятые обозначения: PR – программа ведомого компьютера; IC – внутреннее кодирование кадра типа I; ID – декодирование кадра типа I; PC(n) – внутреннее кодирование Y макроблоков кадра типа P для ведомого компьютера с номером n ; PD – декодирование Y макроблоков кадра типа P; V – поиск векторов движения; PS(n) – компенсация движения для Y макроблоков кадров типа P; GOP – группа кадров, содержащая один кадр типа I и несколько кадров типа P, IPN – групповой адрес ведомых компьютеров, IP(n) – IP-адрес ведомого компьютера с номером n .

Для кодирования видеоконференций в стандарте H.261 при скорости передачи данных в сети 10 Мб/с 50% времени расходуется на передачу данных и 50% – на параллельные вычисления в локальной сети. Отметим, что возможность работы группы ведомых компьютеров по одной программе и использование функции пересылки данных по групповому адресу позволило уменьшить время передачи данных в два раза и обеспечить реальное время для кодирования видеоконференций.

РАСПРЕДЕЛЕННЫЕ ВЫЧИСЛЕНИЯ ДЛЯ СЖАТИЯ ЦИФРОВОГО ВИДЕО В СТАНДАРТЕ MPEG

Для кодирования сжатия цифрового видео в стандарте MPEG [4] необходимо решить проблему скорости обработки, которая составляет $10^{10} \dots 10^{14}$ операций/с. Стандарт MPEG отличается разнообразием видеоформатов и окон поиска, а также тем, что помимо кадров типов I и P в сжатой последовательности содержатся кадры типа B, в которых операция поиска векторов движения выполняется относительно двух опорных кадров типов I и P.

В работе [5] исследованы типы параллелизма, которые можно применять для кодирования сжатия цифрового видео в стандарте MPEG. В таблице приведено количество ведомых компьютеров, которое необходимо при сжатии цифрового видео в реальном времени, выраженное через операцию типа сложение двух операндов для различных видеоформатов и размеров окон поиска. Значения

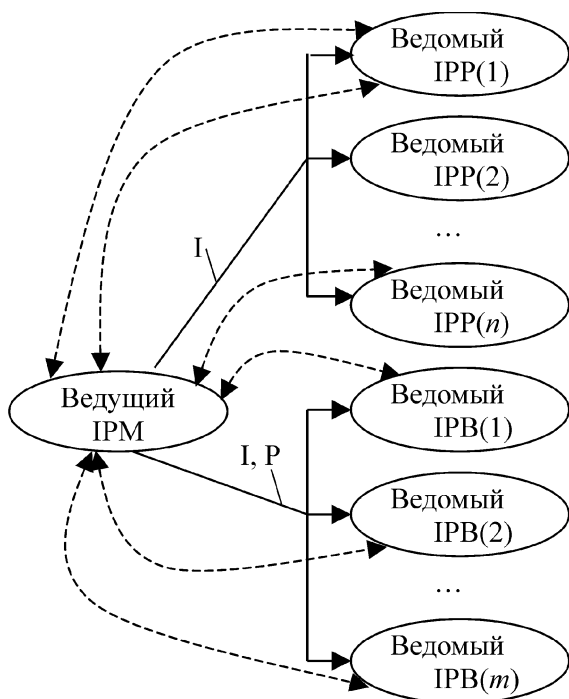


Рис. 5. Организация распределенных вычислений для стандарта MPEG

получены при следующих значениях параметров: время выполнения одной операции над восьмизначными операндами $0,25 \cdot 10^{-9}$ с, частота кадров 30 с^{-1} .

При организации распределенных вычислений целесообразна параллельная обработка многих макроблоков текущих кадров типа В, расположенных между опорными кадрами, и многих макроблоков одного кадра типа Р. При этом один макроблок одного текущего кадра может обрабатываться многими ведомыми компьютерами, и в каждом ведомом компьютере производится сравнение одного макроблока текущего кадра с частью окна поиска. Параллелизма, присущего задаче, при нали-

Необходимое количество ведомых компьютеров для кодирования сжатия цифрового видео в стандарте MPEG

Видеоформаты, размер кадра $A \times C$	Параметр окна поиска (в количестве макроблоков)				
	1	2	4	8	16
SIF 352×240	2	4	24	100	370
CCIR 601 720×486	7	26	102	412	1651
EDTV 960×486	9	35	137	550	2201
HDTV 1920×1080	38	152	608	2446	9784

чи неограниченного числа компьютеров в сети достаточно для организации сжатия цифрового видео в реальном времени.

Таким образом, в данном алгоритме распараллеливания будут задействованы один ведущий компьютер IPM и два типа ведомых, обрабатывающих кадры типа В – IPB(m) и кадры типа Р – IPP(n) (рис. 5). Необходимое число ведомых компьютеров зависит от используемого видеоформата (размера кадра) и окна поиска.

Кодирование в реальном времени предполагает обмен между компьютерами со скоростью 100 Мб/с для несжатых кадров или со скоростью 10 Мб/с для сжатых с помощью внутреннего кодирования кадров. Если скорость передачи данных в сети ниже, то сжатие в реальном времени невозможно.

ЗАКЛЮЧЕНИЕ

В настоящее время кодирование сжатия цифрового видео на персональном компьютере в реальном времени не производится. Для сжатия применяются или супер-ЭВМ, или специализированные параллельные процессоры высокой производительности. Распределенные вычисления в сетях из недорогих компьютеров позволят решать задачи сжатия цифрового видео в реальном времени.

В рассмотренном случае для эффективного управления распределенными вычислениями в Сети при сжатии цифрового видео существенно использование функции передачи данных по групповому адресу, позволяющее уменьшить время передачи данных и ускорить общее время решения задачи. Передачу данных по групповому адресу в сети Интернет можно обеспечить в соответствии с шестой версией протокола TCP/IP.

Анализ предложенного способа управления сетевыми ресурсами для задач сжатия цифрового видео показывает возможности эффективного применения языка ПАРСЕК к решению распределенных задач реального времени.

ЛИТЕРАТУРА

1. Foster I., Kesselman C., Tuecke S. The Anatomy of the Grid. Enabling Scalable Virtual Organizations. <http://www.globus.org/research/papers/anatomy.pdf>.
2. Затуливетер Ю.С. К единому полю компьютерной информации в концепции исчисления древовидных структур // Тр. Ин-та пробл. управл. РАН. – 2002. – Т. XVIII. – С.113–126.
3. Затуливетер Ю.С., Халатян Т.Г. ПАРСЕК – язык компьютерного исчисления древовидных структур с открытой интерпретацией. Стендовый вариант системы программирования. – М.: Ин-т пробл. управл. РАН, 1997.
4. Фищенко Е.А. Сжатие цифрового видео: методы и стандарты // Датчики и системы. – 2002. – № 8. – С.48–63.
5. Затуливетер Ю.С., Фищенко Е.А., Сухов Е.Г. Метод оптимизации параллельного процессора для сжатия цифрового видео // Тр. Ин-та пробл. управл. РАН. – 2002. – Т. XVIII. – С.94–112.

☎ (095) 334-92-09

E-mail: zvt@ipu.rssi.ru, fish@ipu.rssi.ru

