

Кортаев Кирилл Сергеевич

Математическая модель планирования групп процессов
с гарантированным качеством обслуживания

Специальность 05.13.18 —
Математическое моделирование, численные методы
и комплексы программ

АВТОРЕФЕРАТ
диссертации на соискание учёной степени
кандидата физико-математических наук

Москва — 2006

Работа выполнена на кафедре информатики Московского физико-технического института (государственного университета).

Научный руководитель: кандидат физико-математических наук
ТОРМАСОВ Александр Геннадьевич

Официальные оппоненты: доктор технических наук
профессор
СЕМЕНИХИН Сергей Владимирович
кандидат физико-математических наук
ОБЕРНИХИН Виталий Александрович

Ведущая организация: Институт Автоматизации
Проектирования РАН

Защита состоится «_____» _____ 2006 года в _____ часов на заседании диссертационного совета К 212.156.02 при Московском физико-техническом институте (государственном университете) по адресу: 141700, Московская область, г. Долгопрудный, Институтский пер., д. 9.

С диссертацией можно ознакомиться в библиотеке Московского физико-технического института (государственного университета).

Автореферат разослан «_____» _____ 2006 года.

Учёный секретарь диссертационного совета К 212.156.02,
кандидат физико-математических наук

Федько О. С.

Общая характеристика работы

Актуальность темы

В диссертационной работе рассматривается математическая задача планирования и распределения вычислительных ресурсов операционной системы (процессорного времени) между группами процессов (пользователями) с учетом следующих требований: пропорциональное распределение ресурсов во времени и гарантированное качество обслуживания; возможность ограничения группы заданной долей ресурса; корректная работа на многопроцессорных системах. Эта задача актуальна для современных систем разделения пользователей и их изоляции.

Особенно остро задача разделения ресурсов встала в связи с бурным развитием систем виртуализации и изоляции программного обеспечения и операционных систем (ОС). Виртуализация позволяет запускать на одном физическом сервере несколько виртуальных машин (ВМ) или виртуальных серверов (ВС), и, таким образом, позволяет, например, программное обеспечение с нескольких машин выполнять на одном физическом компьютере более экономно используя имеющиеся мощности. По отчетам аналитических агентств IDC и Gartner рынок систем виртуализации растет огромными темпами, и ожидается, что за 2006 год рост составит более 100%. Начиная с 2007 года компания Intel планирует оснащать все свои настольные процессоры технологией VT для аппаратной поддержки виртуализации. Аппаратная виртуализация повысит производительность систем виртуализации, и ожидается, что в скором времени каждый настольный компьютер уже изначально будет поставляться с какой-либо системой виртуализации, позволяющей запускать приложения в своих собственных виртуальных средах. При этом при виртуализации остро встают вопросы распределения ресурсов, их планирования и гарантирования. Процессорное время, как ресурс, рассматриваемый в данной работе – не исключение.

Помимо систем виртуализации, вопрос разделения, планирования и гарантирования ресурсов возникает и в самих современных многопользовательских ОС. В ОС, таких как Windows и Linux, которые позволяют сотням пользователей одновременно работать и выполнять свои приложения, до сих пор нет адекватных средств для управления ресурсами: памятью, процессором, диском и др. В лучшем случае эта задача решается на уровне *нитей* или *процессов*, что никак не позволяет контролировать *конечных пользователей*. Помимо этого проблема заключается в тонком различии между всеми этими ресурсами, что не позволяет эффективно решить задачу единым подходом для всех видов ресурсов сразу.

Математическим моделям планирования ресурсов посвящено большое количество работ за рубежом. В последнее десятилетие тема активно развивается и в нашей стране.

Цель работы

Целью работы является создание для ОС общего назначения (Windows, Linux) единой математической модели планирования процессорного времени со следующими специальными свойствами:

- обеспечение минимально гарантированной доли процессорного времени группе процессов,
- обеспечение ограничения на максимально потребляемое группой процессов время.

Исследуется возможность распределения процессорного времени на базе существующих моделей с использованием виртуального времени и вектора ошибок.

Для анализа качества обслуживания планировщиков предложены методы оценки математических моделей на вносимую задержку в обслуживании и удовлетворение критерия пропорциональности распределения.

Научная новизна

1. Предложена математическая модель пропорционального планирования *групп* процессов, позволяющая управлять распределением процессорного времени. Данная модель *гарантирует* выделение указанной доли времени группе процессов в соответствии с оценками, данными в работе.
2. В дополнение к предыдущей модели, обеспечивающей гарантированную долю процессорного времени, предложена математическая модель и алгоритм, обеспечивающий в рамках этой модели жесткие пределы на потребляемые группой процессов ресурсы. Вместе эти модели позволяют задавать верхний и нижний пределы потребляемых ресурсов.
3. Для внедрения в стандартный планировщик операционной системы указанных моделей планирования разработана и обоснована модель “виртуальных процессоров”.
4. На основании предложенных математических моделей разработан соответствующий комплекс программ. Проведены численные эксперименты с его использованием. Проведено сравнение различных планировщиков задач; продемонстрированы особенности многопроцессорных систем с точки зрения планирования процессорных ресурсов.

Научная и практическая ценность работы

На примере разработанного комплекса программ продемонстрирована принципиальная пригодность предложенной модели пропорционального планирования групп процессов и модели жестких ресурсных ограничений, включая требуемую корректную поддержку многопроцессорных систем.

Созданный комплекс программ для ЭВМ может служить основой для разработки моделей изоляции пользователей и программ, обеспечиваю-

щих количественные и качественные гарантии в обслуживании. Реализация данных моделей внедрена как составная часть в продукты Virtuozzo¹ и OpenVZ², и ее использование показывает, что модели представляют не только научный интерес и задают новые направления в теории планирования процессорных ресурсов, но и успешно решают поставленные перед ними задачи на практике, а именно позволяют управлять и контролировать потребление процессорных ресурсов ВС. Вместе данные программные продукты установлены на более чем 8,000 серверов по всему миру и обеспечивают корректную работу и изоляцию более чем 400,000 виртуальных серверов.

Разработанные в ходе исследования методы планирования могут быть также адаптированы для управления потоком сетевых данных и дисковым вводом-выводом в случае нескольких устройств — сетевых каналов или дисков (аналог многопроцессорности).

Методы исследования

Для построения основных моделей работы используются теория алгоритмов, методы теории операционных систем и системного программирования, математические методы анализа. Широко используется аппарат математического анализа и математических моделей вычислительных систем.

Для проведения численных исследований предложенные модели реализованы как составная часть комплекса программ, и проведены эксперименты с использованием модельных и реальных приложений.

Аппробация результатов работы

Материалы, отражающие содержание диссертационной работы, опубликованы в работах [1–12]. В опубликованных работах автору принад-

¹<http://www.virtuozzo.com>

²<http://openvz.org>

лежит более 60% материала, связанного с изложением математических моделей управления ресурсами, соответствующими алгоритмами и комплексами программ. Результаты диссертации обсуждены и одобрены специалистами на следующих научных конференциях:

- XLVII и XLVIII научной конференциях МФТИ «Современные проблемы фундаментальных и прикладных наук» (Долгопрудный, 2004, 2005),
- международной конференции «Cluster 2005» (Boston, USA, 2005),
- X всероссийской научно-практической конференции «Научное творчество молодежи» (Кемерово, 2006),
- всероссийской научно-практической конференции посвященной 60-летию ТОИПКРО (Томск, 2006),
- III международной конференции разработчиков свободных программ на Протве (Обнинск, 2006).

Также основные результаты были доложены и получили одобрение специалистов на следующих научных семинарах:

- Linux Kernel Summit (Ottawa, Canada, 2006),
- LinuxTag (Wiesbaden, Germany, 2006),
- кафедры информатики Московского физико-технического института (государственного университета) (2003-2006гг).

Структура и объём диссертации

Диссертация состоит из введения, четырех глав, заключения и списка использованных источников, включающего 107 наименований. Диссертация изложена на 111 страницах, включает 2 таблицы и 19 рисунков.

Содержание работы

Во введении обосновывается актуальность темы исследования, научная и практическая ценность работы, её научная новизна, кратко излагается содержание и структура диссертации.

В первой главе даётся развёрнутый анализ того, какие требования предъявляются к современным планировщикам процессов. Основные из которых: возможность планирования групп процессов (пользователей, а не одиночных процессов), пропорциональное распределение процессорного времени между группами, обеспечение жестких пределов на потребляемые ресурсы, корректная работа на многопроцессорных (SMP) системах. А в качестве важных параметров качества обслуживания вводится понятие «задержки в обслуживании» (latency) и сложности алгоритма. Приводятся математические оценки задержки и пропорциональности, полученные для идеального планировщика.

Далее в главе дается обзор исследований по теме диссертации и современного состояния проблемы в соответствие с сформулированными требованиями.

Во второй главе рассматривается математическая модель пропорционального планирования процессорных ресурсов. Автором предложена модель называемая PSFQ и являющаяся естественным обобщением SFQ модели, разработанной P. Goyal и H. Vin³, на случай SMP планирования групп процессов. Модель вводит два основных отличия от оригинальной модели, первоначально разработанной для планирования сетевых данных: каждый процессор подобен своему каналу передачи данных с фиксированной пропускной способностью и для SMP систем существуют такие веса групп, что условие пропорциональности может нарушаться (недопустимые веса, not feasible).

³Goyal P., Vin H. M., Cheng H. Start-Time Fair Queuing: A Scheduling Algorithm for Integrated Services Packet Switching Networks // Proceedings of ACM SIGCOMM'96. — 1996. — Pp. 157–168.

Обозначим за g_i группу процессов, а за w_i – ее вес (приоритет), где $i \in \{1, n\}$, n – общее число групп. Пусть также $W_i(t_1, t_2)$ есть работа, совершенная группой g_i , или количество процессорного времени, выделенного группе за интервал времени $[t_1, t_2]$. Тогда для идеально пропорционального планировщика процессов каждая группа на некотором интервале времени должна была бы получать свою долю B_i от общего процессорного времени B , равную:

$$B_i = B \times \left(\frac{w_i}{\sum_{j=1}^n w_j} \right)$$

Т.е. выполнялось бы равенство:

$$\frac{W_i(t_1, t_2)}{w_i} - \frac{W_j(t_1, t_2)}{w_j} = 0 \text{ для } \forall i, j \quad (1)$$

и в простейшем случае $W_i(t_1, t_2) = w_i(t_2 - t_1)$.

Данное условие, однако, не выполнимо для реальных систем, т.к. не учитывает тот факт, что процессы в системе рождаются и умирают, просыпаются и засыпают, а процессорное время в ОС дискретно – существует минимальный квант времени q , в течение которого процесс не может быть вытеснен. Поэтому пропорциональным планировщиком называется тот, который делает это выражение как можно ближе к нулю.

SFQ модель, как и многие другие модели пропорционального планирования, основывается на модели виртуального времени⁴, которое вводится как функция $v(t)$, меняющаяся со скоростью равной скорости изменения функции $W_k(t)/w_k$ обслуживаемого потока данных, т.е.

$$\frac{d}{dt}v(t) \triangleq \frac{d}{dt} \frac{W_k(t)}{w_k}.$$

Можно показать, что такая функция $v(t)$ является кусочно-гладкой с производной:

$$\frac{dv(t)}{dt} = C \cdot \left(\sum_{k \in B(t)} r_k \right)^{-1}, \quad (2)$$

⁴Zhang L. Virtual clock: A new traffic control algorithm for packet switching // *ACM Transactions on Computer Systems*. – May 1991. – no. 9(2). – Pp. 101–124.

которая меняется каждый раз при изменении множества активных сессий $B(t)$. Функция $v(t)$ может рассматриваться как некоторое нормализованное количество сервиса, выдаваемое системой обслуживаемым сессиям. Данная функция затем используется для сравнения выданного количества сервиса k -той сессии. Обслуживается в первую очередь та сессия, которая больше всего недополучила сервиса по сравнению с $v(t)$. Вычисление $v(t)$ прямым способом не эффективно, т.к. требует постоянного пересчета суммы весов в выражении (2).

SFQ модель, в качестве базовой, в работе была выбрана не случайно. Во-первых, SFQ модель позволяет вычислять $v(t)$ за время сравнимое с $O(1)$. Во-вторых, применительно к сетевым потокам данных, SFQ модель предлагает качество обслуживания близкое к теоретическому:

Теорема 1 (О наилучшей пропорциональности планирования⁵). *Рассмотрим класс S планировщиков пакетов данных таких, что для любого интервала времени (t_1, t_2) и любой пары активных потоков данных k и j выполняется условие:*

$$\left| \frac{W_k(t_1, t_2)}{w_k} - \frac{W_j(t_1, t_2)}{w_j} \right| \leq D(k, j), \quad (3)$$

где $D(k, j)$ не зависит от временного интервала (t_1, t_2) . Тогда для любого планировщика класса S будет выполняться неравенство:

$$D(k, j) \geq \frac{1}{2} \left(\frac{L_k^{\max}}{w_k} + \frac{L_j^{\max}}{w_j} \right), \quad (4)$$

где L_k^{\max} – максимальная длина пакета потока k .

Стандартный SFQ алгоритм выбирает на исполнение единичные процессы. Группа процессов отличается в этом плане от единичного процесса тем, что может исполняться сразу на нескольких процессорах. При этом

⁵Golestani S. J. A self-clocked fair queueing scheme for broadband applications. // Proceedings of IEEE INFOCOM'94. — Jun 1994. — Pp. 636–646.

стандартный SFQ алгоритм никак этого не учитывает. Покажем, как можно его расширить, чтобы он мог использоваться не только для планирования процессов, но и групп процессов.

Для этого каждой группе процессов припишем два новых параметра:

- $runnable_i$ — максимальное количество процессоров, которое данная группа g_i может занять. В простейшем случае, это количество готовых к исполнению процессов в группе;
- $running_i$ — количество выделенных группе g_i процессоров в данный момент времени.

Существенным здесь является новое ограничение на планирование: $running_i \leq \min(N, runnable_i)$, где N — количество процессоров в системе.

Вес группы w_g называется допустимым (feasible) в том, и только в том случае, если:

$$\frac{w_g}{\sum_{i \in R} w_i} \leq \frac{runnable_g}{N}, \quad (5)$$

где R — множество готовых к исполнению групп. Иными словами, вес является допустимым, если группа в состоянии потреблять назначенную ей долю процессорного времени.

В работе показывается, что в случае недопустимых весов нарушается ограниченность роста виртуального времени групп относительно друг друга, что приводит к нарушениям пропорциональности планирования. Получена оценка максимального роста виртуального времени ΔS_{max} , в соответствие с которой модифицируется исходная модель.

Полная PSFQ модель планирования групп процессов, полученная в работе и учитывающая особенности многопроцессорных систем, выглядит следующим образом:

1. В момент времени t , когда группе требуется выдать квант времени q ,

ее метка старта вычисляется как

$$S_g = \min\left(\max(v(t), F_g), v(t) + \Delta S_{max}\right), \quad (6)$$

где

$$\Delta S_{max} = q \cdot \frac{N}{\min_k(w_k)}. \quad (7)$$

При этом значение $running_g$ увеличивается на 1.

2. По истечению кванта времени или когда группа освобождает процессор добровольно, метка окончания вычисляется как

$$F_g = S_g + \frac{W(t_1, t_2)}{w_g}, \quad (8)$$

где $[t_1, t_2]$ – последний промежуток исполнения группы i , соответственно, совершенная на этом промежутке времени работа $W(t_1, t_2)$.

При этом значение $running_g$ уменьшается на 1.

3. Виртуальное время в системе $v(t)$ вычисляется в зависимости от того, есть ли выполняющиеся задания на процессоре:

$$v(t) = \begin{cases} \min(S_g) & \text{если есть исполняющаяся группа} \\ \max(F_g) & \text{если процессоры простаивают (idle)} \end{cases} \quad (9)$$

4. Изначально $v(0) = 0$ и $S_g = F_g = 0$
5. Каждый раз, когда в группе просыпается или засыпает процесс, значение параметра $runnable_i$ увеличивается или уменьшается на 1 соответственно.
6. Каждый раз, когда нужно принять решение о планировании, выбирается группа с минимальной меткой старта S_g , удовлетворяющая условию:

$$running_g < \min(runnable_g, N) \quad (10)$$

В работе доказан ряд теорем относительно свойств полученной модели. В частности:

Теорема 2 (Оценка пропорциональности для допустимых весов). *Для любого временного интервала $[t_1, t_2]$, на котором две группы f и g готовы к исполнению в течение всего интервала, в случае допустимых весов групп выполняется неравенство, устанавливающее следующую максимальную разницу в получаемых группами ресурсах:*

$$\left| \frac{W_f(t_1, t_2)}{w_f} - \frac{W_g(t_1, t_2)}{w_g} \right| \leq qN \left(\frac{1}{w_f} + \frac{1}{w_g} \right) \quad (11)$$

Данная теорема аналогична теореме 1 и показывает связь между длиной пакета в сетевых алгоритмах и величиной кванта планировщика процессорного времени.

Теорема 3 (Максимальная задержка в обслуживании). *Для PSFQ модели планирования максимальная задержка (latency) в обслуживании группы g не превосходит:*

$$latency \leq q \cdot \max \left[1, N \left(\frac{\sum_{k \in R} (w_k)}{w'_g} - 1 \right) \right], \quad (12)$$

где R – множество готовых к исполнению групп, а $w'_g = w_g$ в случае допустимых весов и

$$w'_g = \frac{runnable_g}{N} \sum_{i \in R} w_i,$$

в случае недопустимых весов.

Теорема 4. *В PSFQ модели для исполняющейся за любой интервал времени $[t_1, t_2]$ группы f с недопустимым весом w_f выполняется следующее неравенство:*

$$W_f(t_1, t_2) \geq \frac{runnable_g}{N} (t_2 - t_1) - latency \quad (13)$$

Теорема 5 (Сложность PSFQ модели). *Математическая сложность PSFQ модели не превосходит $O(\ln n)$ при операциях планирования, просыпания и засыпания группы и $O(n)$ при создании/удалении группы.*

В третьей главе предлагается и анализируется разработанная автором модель жесткого ограничения потребляемого группой процессов процессорного времени. Модель пропорционального планирования, рассматриваемая в предыдущей главе, обеспечивает гарантии на минимально выделяемое группе процессов процессорное время, а именно:

$$W_g(t_1, t_2) \geq \frac{w_g}{\sum_{i \in R} w_i} (t_2 - t_1) - \delta,$$

где $0 \leq \delta \leq \delta_0$ для некоторого произвольного фиксированного δ_0 . Так происходит из-за того, что в реальности зачастую не все группы готовы исполняться:

$$W_g(t_1, t_2) \geq \frac{w_g}{\sum_{i \in R} w_i} (t_2 - t_1) - \delta \geq \frac{w_g}{\sum_i w_i} (t_2 - t_1) - \delta,$$

где множество R – множество готовых к исполнению групп.

Поэтому, для предоставления помимо гарантий еще и верхней границы потребляемого группой g процессорного времени, требуется так же модель ограничения ресурсов, такая что:

$$W_g(t_1, t_2) \leq r_g(t_2 - t_1), \tag{14}$$

где $r_g \in [0, 1]$ – максимальная доля всех имеющихся процессорных ресурсов.

Для этого в модели вводится понятие кредита процессорного времени c_g , который позволено выделить данной группе, и, соответственно, модель называется “кредитной”. Т.к. минимальный квант времени ОС q , то в случае $c_g < q$ группа не должна получать процессорное время до тех пор, пока ей не выделят следующий кредит в момент времени d_g , и кредит c_g снова не станет больше q . Первоначальный кредит $c_g = q \cdot N$. Это позволяет группе сразу перейти в состояние исполнения при пробуждении и

избежать ненужной начальной задержки в обслуживании, что важно для интерактивных приложений.

В работе также предлагается оценка времени d_g – момента времени t , до которого группа g не должна получать процессорное время в случае, если ее кредит исчерпан. Это существенно, т.к. позволяет снизить вычислительные затраты на модельные расчеты. А условие (7) PSFQ модели гарантирует корректную работу этих двух моделей вместе, ведь ограничение группы может приводить к недопустимым весам с точки зрения пропорционального планирования.

При переходе из состояния выполнения в момент времени t_2 в любое другое состояние, нужно увеличить кредит c_g в соответствие с прошедшим временем (с момента предыдущего перехода t_1), плюс нужно вернуть неиспользованную часть кванта времени $\Delta t = q - (t_2 - t_1)$ (которая сразу же целиком засчитывается при выдаче процессорного времени):

$$c_g(t_2) = \min(c_g(t_1) + r_g N(t_2 - t_1) + [q - (t_2 - t_1)], qN) \quad (15)$$

В случае, если группа исчерпала свой кредит и не может больше исполняться, то она переводится в новое состояние *delayed*, в котором она никогда не получает процессорное время. Выйти из этого состояния группа сможет в момент времени d_g , когда у нее накопиться достаточно кредита, чтобы исполняться на всех процессорах одновременно:

$$d_g = t_2 + \frac{q \cdot \text{runnable}_g - c_g(t_2)}{r_g N} \quad (16)$$

В работе доказан ряд теорем относительно свойств полученной модели. В частности:

Теорема 6. *Для любой группы g и интервала $[t_1, t_2]$ кредитная модель ограничивает потребляемое группой количество ресурса $W_g(t_1, t_2)$ величиной:*

$$W_g(t_1, t_2) \leq r_g N(t_2 - t_1) + qN$$

Причем это наименьшая возможная оценка сверху.

Не смотря на то, что данная оценка отличается от исходного условия (14), показано, что дополнительное слагаемое $+qN$ вызвано дискретностью модели и чрезвычайно важно для планирования интерактивных приложений.

Дополнительно показано, что предложенная модель имеет наименьшую теоретически возможную задержку в обслуживании:

Теорема 7. *Вносимая кредитной моделью задержка в обслуживании (latency) группы g не превосходит:*

$$latency \leq q \left(1 + \frac{1}{r_g} \right) \quad (17)$$

Причем это минимальная из возможных оценок сверху.

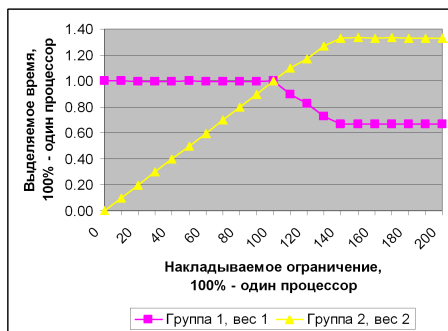
Теорема 8. *Математическая сложность кредитной модели не превосходит $O(\ln n)$, где n – количество групп в состоянии *delayed*.*

В четвертой главе рассматривается модель виртуального процессора. Данная модель позволяет без особых усилий объединить две предложенные выше модели (или любой другой пропорциональный планировщик) со стандартным планировщиком ОС, что позволяет получить комбинированный планировщик, имеющий положительные свойства всех участвующих моделей. Так, использование стандартного планировщика ОС вместе с пропорциональным позволяет:

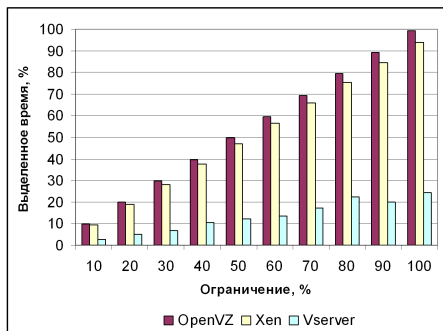
- сохранить семантику ОС, которая предоставляет возможность менять приоритеты отдельных процессов внутри групп,
- получить планировщик, учитывающий интерактивность процессов,
- получить планировщик, учитывающий аппаратные особенности и разнородность аппаратных средств современных многопроцессорных систем (HyperThreading, multicore, NUMA).

Идея состоит в разделении информации об очередях процессов ОС на две части — процессорно-зависимую и процессорно-независимую. Процессорно-зависимая информация содержит в себе данные о состоянии физического процессора, а процессорно-независимая о виртуальном (очереди процессов). Благодаря такому разделению становится возможным исполнять виртуальные процессоры (т.е. процессы различных виртуальных процессоров) на различных физических процессорах. Последнее позволяет устранить зависимость решений планировщиков разных уровней, т.к. после того, как выше предложенный пропорциональный планировщик выбрал группу на исполнение, всегда можно выбрать виртуальный процессор, в котором стандартный планировщик ОС уже точно сможет выбрать процесс.

Сформулированы возникающие в модели ограничения на планирование, а именно то, что выбираться на исполнение должны только виртуальные процессоры, у которых очередь готовых к исполнению процессов не пуста, и что виртуальный процессор не должен выполняться на нескольких физических, равно как и один физический процессор не должен обрабаты-



(а) Пример с двумя группами (по одному процессу) с весами 1 и 2 на двухпроцессорной машине. Вторая группа последовательно ограничивается долей от 0 до 2 процессоров, постепенно вытесняя при этом менее приоритетную группу.



(б) Сравнение точности предложенной модели ограничения группы процессов с двумя другими моделями, реализованными в проектах Xen и VServer. Группа последовательно ограничивается долей от 0 до 2 процессоров.

Рис. 1. Совместная работа 3-х предложенных моделей.

вать более одного виртуального.

На рис. 1 приведены некоторые результаты численных экспериментов на SMP системе, демонстрирующие: а) корректную работу внедренных в стандартный планировщик ОС Linux алгоритмов на базе разработанных математических моделей; б) высокую точность предложенной модели ограничения потребляемого группой процессов процессорного времени по сравнению с двумя другими доступными реализациями.

В заключении указаны основные результаты и выводы диссертации.

Основные результаты и выводы диссертации

1. Предложена математическая модель пропорционального планирования групп процессов, позволяющая управлять распределением процессорного времени. Данная модель гарантирует выделение указанной доли времени группе процессов и включает математические оценки точности пропорциональности и задержки в обслуживании.
2. Предложена математическая модель ограничения потребляемого группой процессов процессорного времени. На основании данной модели получены оценки качества обслуживания.
3. Разработана и реализована модель “виртуальных процессоров”, позволяющая объединить планировщик групп процессов со стандартным планировщиком ОС.
4. Предложенные модели реализованы в виде комплекса программ, обеспечивающего количественные и качественные гарантии в обслуживании. Проведены численные исследования свойств предложенных моделей на многопроцессорных системах.

Список публикаций по теме диссертации

1. *Коротаев К., Емельянов П.* Многоуровневый планировщик процессорного времени для групп процессов, обеспечивающий гарантии в обслуживании // *Информационные технологии.* — М. 2006. — № 6. — С. 58–63.
2. *Коротаев К., Савочкин А.* Концепция виртуальных процессоров для планировщика задач, реализующего изоляцию групп процессов // *Процессы и методы обработки информации: Сб.ст.* — М.: Моск. физ.-техн. ин-т, 2005. — С. 81–87.
3. *Емельянов П., Коротаев К., Луковников И.* Основные проблемы реализации алгоритмов пропорционального планирования // *Процессы и методы обработки информации: Сб.ст.* — М.: Моск. физ.-техн. ин-т, 2006. — С. 86–91.
4. Основные проблемы обеспечения точности управления при наложенном управлении ресурсами в современных операционных системах / А. Тормасов, П. Емельянов, К. Коротаев и др. // *Процессы и методы обработки информации: Сб.ст.* — М.: Моск. физ.-техн. ин-т, 2006. — С. 237–242.
5. *Коротаев К., Емельянов П.* Модифицированный SFQ алгоритм (MSFQ) для честного распределения процессорного времени на многопроцессорных системах // *Проблемы вычислительной математики, математического моделирования и информатики: Сб.н.тр.* — М.: МЗ Пресс, 2006. — С. 214–233.
6. *Коротаев К., Емельянов П.* Пути адаптации современных групповых планировщиков задач для работы в многопроцессорных системах // *Современные проблемы фундаментальных и прикладных наук. Часть*

- VII. Управление и прикладная математика: Труды XLVIII научной конференции. — М. — Долгопрудный: Моск. физ.-техн. ин-т, 2005. — С. 237.
7. *Коротаев К., Савочкин А.* Модель группового планировщика процессов с использованием виртуальных процессоров // Современные проблемы фундаментальных и прикладных наук. Часть VII. Управление и прикладная математика: Труды XLVIII научной конференции. — М. — Долгопрудный: Моск. физ.-техн. ин-т, 2005. — С. 39.
 8. *Коротаев К., Емельянов П.* Изучение и оптимизация производительности групповых планировщиков процессов на примере честного планировщика ядра ОС Linux // Современные проблемы фундаментальных и прикладных наук. Часть VII. Управление и прикладная математика: Труды XLVIII научной конференции. — М. — Долгопрудный: Моск. физ.-техн. ин-т, 2005. — С. 40.
 9. *Korotaev K.* Hierarchical CPU Schedulers for Multiprocessor Systems, Fair CPU scheduling and Processes Isolation // Proceedings of Cluster'05. — Boston: 2005. — Pp. 53–54.
 10. *Коротаев К., Луковников В., Луковников И.* О существующих планировщиках задач, пропорциональности распределяемого ими процессорного времени и способах улучшения качества обслуживания // Научное творчество молодежи. Часть I. Материалы X Всероссийской научно-практической конференции. — Кемерово: Кемеровский гос. универ-т, 2005. — С. 74.
 11. “Честный” планировщик задач на основе аппроксимации идеального времени (ITAFS): новый подкласс пропорционально-долевых алгоритмов планирования задач для процессоров / А. Костюшко, К. Коротаев, В. Луковников, И. Луковников // Научное творчество молодежи. Часть

I. Материалы X Всероссийской научно-практической конференции. — Кемерово: Кемеровский гос. универ-т, 2005. — С. 75.

12. *Кортаев К. С., Луковников В. В., Луковников И. В.* Многоуровневый планировщик процессов для изоляции процессов и групп пользователей. Проблемы балансировки нагрузки и пути их преодоления // Приоритетные направления модернизации общего образования. Материалы Всероссийской научно-практической конференции, посвященной 60-летию ТОИПКРО. — Т. 1. — Томск: ТОИПКРО, 2006. — С. 174–175.

Коротаев Кирилл Сергеевич

Математическая модель планирования групп процессов с
гарантированным качеством обслуживания

Автореферат

Подписано в печать 12.09.2006

Усл. печ. л. 1.5. Тираж 80 экз. Заказ № 471

Московский физико-технический институт
(государственный университет)

Печать на аппаратуре Rex-Rotary Copy Printer 1280. НИЧ МФТИ