РОССИЙСКАЯ АКАДЕМИЯ НАУК
ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР

СООБЩЕНИЯ ПО ПРИКЛАДНОЙ МАТЕМАТИКЕ

П.Е. ГОЛОСОВ, М.В. КОЗЛОВ, Ю.Е.МАЛАШЕНКО,
И.А. НАЗАРОВА, А.Ф. РОНЖИН

МОДЕЛЬ СИСТЕМЫ УПРАВЛЕНИЯ СПЕЦИАЛИЗИРОВАННЫМ ВЫЧИСЛИТЕЛЬНЫМ КОМПЛЕКСОМ

ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР РАН
МОСКВА 2010

УДК 519.85

Ответственный редактор
член-корр. РАН, доктор физ.-матем. наук
Ю.А. Флеров

Рассмотрена проблема управления заданиями в системе взаимозависимых исполняющих вычислительных устройств. Проблема сводится к задаче выбора стратегии управления выполнением заданий. Для ее решения предложено использовать имитационную модель. В работе подробно описаны результаты численных экспериментов, проиллюстрированные графиками и таблицами.

Ключевые слова: распределенные вычисления, математическое моделирование, теория расписаний, порядковые статистики.

Публикуется в авторской редакции.

Рецензенты: М.Г. Коновалов,
А.В. Лотов

Научное издание

Введение.

Проблема составления расписаний для реально существующих больших вычислительных систем является одной из самых актуальных. В настоящее время предпринимаются попытки решать ее как на аппаратном уровне: автоматическое распределение процессоров, оперативной памяти и каналов связи для передачи данных между составными частями системы, своевременные ответы управляющих блоков (или серверов) на запросы исполняющих устройств и т.д.; так и на диспетчерском: распределение машинного времени, установление порядка выполнения поступивших задач, и др. В любом случае составление расписаний или установление порядка выполнения поступивших задач предполагает оптимальное распределение между ними ресурсов и времени для того, чтобы повысить эффективность функционирования всего вычислительного комплекса.

В данной работе рассматривается диспетчерский уровень управления процессом вычислений, то есть исследуется вопрос о том, как порядок выполнения заданий и распределение ресурсов влияет на конечный результат. В реальной ситуации диспетчер имеет дело с неопределенностью, связанной как с составом заданий, так и со временем их счета. Кроме того, пользователь может назначать директивное время решения задач достаточно произвольно, пытаясь оказать косвенное "давление" на диспетчера, ставя своей целью получить приоритет в обслуживании.

В силу очевидных причин нельзя априори гарантировать, что все требования, наложенные на процесс функционирования системы и предъявляемые к ходу выполнения (решения) задач (заданий), будут приняты во внимание в готовом расписании. Для конкретного вычислительного комплекса может не существовать "графика выполнения"(допустимого упорядочивания), учитывающего всю совокупность ограничений и условий для всех задач. Однако в данной ситуации наличие плана лучше, чем его отсутствие, и "разумные" диспетчерские правила, безусловно, помогут удовлетворить имеющиеся запросы. Здесь речь может идти о составлении допустимого расписания, по-возможности отвечающего большей части требований. При наличии нескольких таких расписаний ответ на вопрос о том, какое из них в том или ином смысле "лучше может дать вычислительный эксперимент.

Данная работа является первой и в некотором смысле постановочной из предполагаемого цикла работ, посвященных построению расписания для определенного класса задач, выполняемых в системе, состоящей из нескольких параллельных машин, и организована следующим образом. В §1 дается неформальное описание исследуемой задачи и указываются общие методы ее решения. В §2, §3 формально описывается предлагаемая математическая модель и исследуются ее свойства. В §4 обсуждаются результаты вычислительного эксперимента.

§1. Общее описание модели

Рассмотрим модель гипотетической информационно-вычислительной системы, основными элементами которой являются вычислительный ресурс, потребители ресурса и "диспетчер управляющий процессом выполнения заданий, предъявленных для выполнения.

Вычислительный ресурс будем трактовать как набор компьютеров (процессоров), способных выполнять предписанные операции вычислительного характера, и далее называть исполняющими вычислительными устройствами (ИВУ). Внутренняя конструкция таких устройств не рассматривается, но считается, что производительность (мощность) и объем их памяти известны. Кроме того, в системе существует центральное управляющее вычислительное устройство (ЦУВУ), которое управляет работой всех ИВУ и организует процесс выполнения всех заданий: распределяет задачи (или их части) между доступными ИВУ, а также контролирует ход их выполнения. Управление самим ЦУВУ может осуществляться как автономной программой-"диспетчером так и непосредственно диспетчером-программистом. ЦУВУ может взять для исполнения как одно, так и несколько заданий одновременно, разбить их на произвольное число подзаданий, распределить по соответствующим ИВУ, и начать их выполнение. При этом ход выполнения задания по той или иной причине может быть прерван командой с ЦУВУ.

Предполагается, что на каждом единичном ИВУ в единицу времени может решаться только одна подзадача. Указанная особенность ЦУВУ — способность обрабатывать как одно, так и несколько заданий одновременно, позволяет существенно расширить рамки поиска при составлении расписания.

Далее при исследовании информационно-вычислительной системы будем считать, что все ИВУ надежны и в процессе решения задач из строя не выходят.

Предположим, что пользователи системы предлагают для выполнения задания одного типа. Каждое из них характеризуется предельно допустимым временем пребывания задания в системе ("дедлайном") и максимальным возможным (предельным) числом необходимых "элементарных" вычислительных операций ИВУ, однако время решения каждой конкретной задачи является случайной величиной. Далее предполагается, что время выполнения может составить от нескольких минут до нескольких десятков часов.

Допустимое время пребывания в системе каждого задания определяет пользователь. Эта величина вместе с временем решения задачи, в конечном итоге, оказывает существенное влияние на порядок выполнения задач.

Для простоты предположим, что выполнение и получение решения для каждой из задач может быть описано случайной выборкой без возвращения. Рассмотрим урну с двумя типами шаров: черными и белыми. Определим вытягивание белого шара как успех, черного – как неудачу. Пусть N — число всех шаров в урне, белых шаров — m. Задание считается выполненным, как только из урны извлечен последний белый шар. N определяет объем перебора (максимальное число элементарных операций ИВУ) в худшем случае, т.е. когда m-й белый шар извлекается последним. В общем случае длительность решения такой задачи является случайной величиной.

Рассмотрим ситуацию, когда задания поступают от пользователей одновременно, и каждое из них может быть поставлено на обработку в любое определенное диспетчером время.

Далее будем полагать, что большинство задач имеет решение, и в урне присутствует ровно один белый шар. Задание считается невыполненным, если его дедлайн наступил до извлечения белого шара, а вычислительный ресурс, ушедший на его выполнение, считается потерянным. Далее рассмотрим также случай, когда белый шар в урне отсутствует, что в реальности соответствует сбою в системе, или неправильной постановке задачи. В этой ситуации задание будет считаться успешно выполненным, если полный перебор всех черных шаров будет произведен до наступления дедлайна, иначе — использованный ресурс также уйдет в потери.

Требованием пользователей является выполнение каждой из задач до наступления ее дедлайна.

В обязанности диспетчера входит распределение вычислительного ресурса и формирование очередности выполнения заданий. При этом он может поставить на обработку как одно, так и несколько заданий одновременно. Диспетчер должен учитывать, что каждое из заданий должно быть завершено до наступления дедлайна. Если к указанному моменту соответствующее задание не было закончено, то его выполнение прекращается. В идеале диспетчер должен управлять системой так, чтобы наиболее полно удовлетворить требования пользователей и минимизировать возможный ущерб.

Описанная проблема может быть исследована с помощью моделей и методов теории расписаний. Действительно, рассматриваемая информационно-вычислительная система может быть описана как одностадийная [1], состоящая из нескольких параллельных машин, имеющих в зависимости от директивы диспетчера одинаковую или различную мощность, а упорядочивание заданий — как составление расписания для совокупности различных ИВУ. Заметим, что специфика поступающих к диспетчеру заданий и неопределенность, связанная с длительностью их обработки, приводит к необходимости использования имитационного моделирования и методов теории вероятностей для анализа результатов последнего.

§2. Постановка задачи

Для упорядочивания заданий воспользуемся следующей классической моделью. Пусть имеется множество $Z = \{Z_1, Z_2, ..., Z_m\}$, состоящее из $m \geq 2$ заданий (работ), поступивших в систему одновременно в момент времени $t = 0$, и обслуживающее их одностадийное ЦУВУ, которое может рассматриваться либо как единственное устройство, имеющие производительность W^*, либо как совокупность единичных ИВУ, имеющих различную, назначенную диспетчером производительность, суммарно равную W^*.

Предполагается, что каждое задание характеризуется исходными данными, в том числе объемом s_i^* (максимальным числом вычислительных операций, необходимых для выполнения i-го задания) и директивным сроком окончания его выполнения (дедлайном) $d_i,\ i = \overline{1, m}$.

Известно, что время решения каждого задания является случайной величиной X_i, имеющей функцию распределения $F_i(x)$ на отрезке $[0; \overline{\tau_i}]$, где $\overline{\tau_i}$ — максимально возможное время обработки задания на общем ресурсе системы мощности W^*. Как правило, значение $\overline{\tau_i}$ оценивается через величину s_i^*. Будем считать, что процесс обслуживания может начаться с любого из имеющихся заданий, и перераспределение ресурса производится только в момент окончания обработки одного или нескольких заданий.

По условию диспетчер для выполнения i-го задания выделяет мощность $w_i(0) \leq W^*$, после чего происходит его обработка указанным набором ИВУ. При этом скорость решения отдельной задачи пропорциональна выделенному ей ресурсу. Считается, что ЦУВУ может одновременно обслуживать от одного до $n \leq m$ заданий.

Рассмотрим гипотетическую возможность того, что длительности выполнения каждой задачи из Z оказалась известна заранее (при условии, что для их решения отдается вся мощность, имеющаяся в распоряжении ЦУВУ). Обозначим длительность решения i-й задачи через T_i^W. Согласно теореме 3.2 [2] среднее время пребывания работ в системе $m|1|F$ минимально, если после упорядочивания длительности работ T_i^W не убывают. Такой алгоритм упорядочения называется упорядочением по минимальной длительности работ ($Shortest\ Processing$ – $time\ Sequencing$ (SPT)). Однако в нашем случае в силу специфики задач (момент извлечения белого шара — случайная величина), длительность каждой работы заранее не известна, поэтому, исходя из имеющихся данных о задаче и мощности, которая может быть выделена для её обработки, в каждом отдельном случае оценим максимально возможное время счета до получения результата.

Пусть $A = \{A_1, A_2, ...A_m\}$ — набор априорных оценок длительностей работ множества Z, упорядоченных по возрастанию. Пусть работы из Z производятся согласно имеющейся последовательности A, т.е. первой выполняется работа с оценкой A_1, ..., последней — с оценкой A_m. Если оценка A_i и реальная длительность X_i работы i положительно коррелированы с коэффициентом корреляции 1, то и упорядочение, проведенное на основании оценок, будет соответствовать упорядочению SPT при заранее известных длительностях работ. Для получения оптимального расписания оценки могут быть не идеально точными, они лишь должны отражать характер отношений между длительностями. Если же величины A_i и X_i независимы, то проведенное упорядочение будет случайным.

Для рассматриваемого в данной работе класса задач время выполнения i-й работы является независимой от остальных случайной величиной, поэтому работа с самой короткой длительностью, при любом упорядочении, вообще говоря, может оказаться на произвольном месте. Таким образом, диспетчер не имеет достоверной информации о сроке окончания каждого задания и вынужден принимать решение в условиях неопределенности. Тем не менее, у него всегда есть не менее двух стратегии: имеющееся множество Z, состоящее из m заданий, можно обрабатывать параллельно или последовательно. Рассмотрим обе стратегии более подробно.

§3. Описание регулярных стратегий

Определение 1. Планом выполнения заданий будем называть набор из m векторов $r_i = (r_1^{(i)}, ..., r_m^{(i)}), i = \overline{1, m}$, в котором $r_v^{(1)}, v = \overline{1, m}$ — доля ресурса, выделяемая v-ой задаче при постановке пакета на обработку, а $r_v^{(i+1)}, v = \overline{1, m}$ — доля ресурса, выделяемая v-ой задаче, после того, когда завершено решение i задач, $i = \overline{1, m-1}$.

Обозначим T_i — время пребывания i-й задачи в системе. Если $T_i > d_i$, то задание покидает систему как невыполненное (задача к назначенному сроку не решена). Если $T_i \leq d_i$, то T_i — момент завершения выполнения задания (задача решена в назначенный срок).

В работе [3] с каждой задачей Z_i из Z связывается невозрастающая функция от времени $p_i(t), i = \overline{1, m}$, которая названа функцией полезности или актуальности, и в качестве меры качества плана обработки множества Z предложено использовать математическое ожидание функционала $p(T_1, ..., T_m) = \sum_{i=1}^{m} p_i(T_i)$. В работе [4] предложен план, который максимизирует этот функционал для $m = 2$.

Далее рассмотрим случай симметричной неопределенности, характеризующийся следующими ограничениями:

1) Функции распределения времени решения задачи выбираются случайным образом без повторения из множества $F = \{F_1, ..., F_m\}$ функций распределения. Соответствие задач и функций распределения неизвестно.

2) Функции полезности линейны и равны между собой. Последнее ограничение означает, что задача максимизации математического ожидания функционала $p(T_1, ..., T_m)$ сводится к минимизации математического ожидания среднего времени пребывания в системе $T = \frac{1}{m}(T_1 + ... + T_m)$.

В силу симметрии исходных данных рассматривающиеся ниже планы будем называть регулярными и определять следующим образом. Зафиксируем $k : 1 \leq k \leq m$. Из имеющегося пакета выберем случайным образом k заданий, разделим ресурс на k равных частей и начнем обрабатывать отобранные задачи одновременно. После окончания решения одной или нескольких задач освободившийся ресурс разделим поровну на все незавершенные задачи и продолжим их обработку. Будем продолжать процесс до тех пор, пока все k задач не будут решены. После этого описанная процедура повторяется до тех пор, пока общий список имеющихся задач не будет исчерпан.

Далее рассмотрим два граничных плана — при $k = 1$ и $k = m$, 1-регулярный и m-регулярный — соответственно. При этом 1-регулярный план будем называть последовательным, а m-регулярный — параллельным. Очевидно, что общее время решения всех задач пакета не зависит от плана и равно $X_1 + X_2 + ... + X_m$.

Обозначим через X вектор $(X_1, ..., X_m)$, где X_i — время решения i-го задания — случайная величина, имеющая одну и ту же функцию распределения $F(x)$. Рассмотрим произвольную реализацию X, упорядочим его компоненты по возрастанию и перенумеруем. Полученный вариационный ряд обозначим X^*, для него $X_1^* \leq ... \leq X_m^*$.

Введем следующие обозначения:

$\overline{S}_m = \frac{1}{m}(X_1 + ... + X_m)$ — среднее время решения задач из множества Z;

T_{seq} — среднее время пребывания задачи в системе при последовательном плане;

T_{par} — среднее время пребывания задачи в системе при параллельном плане.

Утверждение 1. Для параллельного плана справедливо следующее равенство:

$$T_{par} = (2m-1)\overline{S}_m - \frac{2}{m}\sum_{k=1}^{m}(k-1)X_k^*.$$

Доказательство. Поскольку при делении ресурса на k частей время решения задач увеличивается в k раз, и освободившийся ресурс делится поровну, справедливы равенства:

$$T_1 = mX_1^*, \;\; T_2 = mX_1^* + (m-1)(X_2^* - X_1^*),$$
$$T_k - T_{k-1} = (m-k+1)(X_k^* - X_{k-1}^*),$$
$$T_k = X_1^* + ... + X_k^* + (m-k)X_k^*, \;\; k = 2, ..., m.$$

Откуда получаем:

$$T_{par} = \frac{1}{m}\sum_{k=1}^{m}[(X_1^* + ... + X_k^*) + (m-k)X_k^*] =$$
$$= \frac{1}{m}(\sum_{k=1}^{m}(m-k+1)X_k^* + (m-k)X_k^*) =$$
$$= \frac{2m-1}{m}\sum_{k=1}^{m}X_k^* - \frac{2}{m}\sum_{k=1}^{m}(k-1)X_k^* =$$
$$= \frac{2m-1}{m}\sum_{k=1}^{m}X_k - \frac{2}{m}\sum_{k=1}^{m}(k-1)X_k^*.$$

Что и требовалось доказать.

Замечание. Пусть время выполнения каждого задания — случайная величина, равномерно распределенная на отрезке [0,1]. Тогда X_k^* имеет бета-распределение с параметрами $(k, m-k+1)$. Для математического ожидания X_k^* справедливо равенство :

$$M\ X_k^* = \frac{k}{m+1}.$$

Отсюда следует, что при использовании параллельного плана математическое ожидание времени пребывания k-го задания в системе равно:

$$\frac{k(2m-k+1)}{2(m+1)}.$$

При этом математическое ожидание времени между выполнением k-го и $(k-1)$-го заданий линейно убывает с ростом k и равно:

$$\frac{m-k+1}{m+1}.$$

Лемма. Пусть случайные величины $X_i, i = \overline{1,m}$, одинаково распределены, независимы, имеют функцию распределения $F(x)$ и первый конечный момент. Тогда справедливо равенство:

$$S(F) = M\sum_{k=1}^{m}(k-1)X_k^* = \left(\begin{array}{c} m \\ 2 \end{array} \right) M\max(X_1, X_2).$$

Доказательство. Согласно формуле (2.1.2) из [5], для функции распределения k-го члена вариационного ряда справедлива следующая формула:

$$P\{X_k^* \le x\} = \sum_{r=k}^{m} \left(\begin{array}{c} m \\ r \end{array} \right) F^r(x)(1-F(x))^{m-r}.$$

Отсюда следует, что

$$\sum_{k=1}^{m}(k-1)MX_k^* =$$

$$= \int_0^1 xd(\sum_{k=1}^{m}(k-1)\sum_{r=k}^{m} \left(\begin{array}{c} m \\ r \end{array} \right) F^r(x)(1-F(x)^{m-r}) =$$

$$= \int_0^1 xd(\sum_{r=1}^{m} \left(\begin{array}{c} m \\ r \end{array} \right) F^r(x)(1-F(x))^{m-r}\sum_{k=1}^{r}(k-1)) =$$

$$= \left(\begin{array}{c} m \\ 2 \end{array} \right)\int_0^1 xd\left(F^2(x)\sum_{r=0}^{m-2} \left(\begin{array}{c} m-2 \\ r \end{array} \right) F^r(x)(1-F(x))^{m-r}\right) =$$

$$= \binom{m}{2} \int_0^1 x d(F^2(x)) = \binom{m}{2} M \max(X_1, X_2).$$

Последнее равенство верно, поскольку $F^2(x)$ является функцией распределения случайной величины $\max(X_1, X_2)$.

Утверждение 2. Для последовательного плана справедливо следующее равенство для условного математического ожидания среднего времени пребывания задачи в системе относительно $\sigma-$алгебры, порожденной случайными величинами $X_1, ..., X_m$:

$$M(T_{seq} \,|X_1\ , ..., X_m) = \frac{m+1}{2} \overline{S}_m.$$

Доказательство. Если задачи из множества Z выполняются последовательно в порядке $i_1, ..., i_m$, то время пребывания k-й задачи в системе равно $T_k = X_{i_1} + ... + X_{i_k}$, а среднее время пребывания задачи в системе равно:

$$T_{seq} = \frac{1}{m} \sum_{k=1}^{m} T_k = \frac{1}{m} \sum_{k=1}^{m} (m-k+1) X_{i_k}.$$

Вероятность выбора каждой из последовательностей $i_1, ..., i_m$ равна $(m!)^{-1}$. Следовательно,

$$M(T_{seq} \,|X_1\ , ..., X_m) =$$

$$= \frac{1}{m} \sum_{k=1}^{m} (m-k+1) \frac{(m-1)!}{m!} (X_1 + ... + X_m) =$$

$$= \frac{1}{m^2} (X_1 + ... + X_m) \sum_{k=1}^{m} (m-k+1) = \frac{m+1}{2} \overline{S}_m.$$

Что и требовалось доказать.

Теорема. Пусть случайные величины $X_1, ..., X_m$ независимы, и множество F состоит из двух функций:

$F_{m-l+1}(x) = ... = F_m(x) = G(x) = 0$, для $x < 1, l = 0, ..., m$, и

$F_1 = ... = F_{m-l}(x) = F(x)$

Для математических ожиданий среднего времени пребывания в системе справедливы равенства:

$$MT_{seq} = \frac{m+1}{2m}(l + (m-l)MX_1),$$

$$MT_{par} = \frac{l^2}{m} +$$

$$+\frac{(m-l)}{m}\left\{(2m-1)MX_1 - (m-l-1)M\max(X_1, X_2)\right\},$$

$$\Delta = M(T_{par} - T_{seq}) = \frac{l}{m}\left(l - \frac{m+1}{2}\right) +$$

$$+\frac{(m-l)}{m}\left\{\frac{3}{2}(m-1)MX_1 - (m-l-1)M\max(X_1, X_2)\right\}.$$

Доказательство следует из утверждений 1, 2 и леммы. В частности, поскольку $X^*_{m-l+1} = ... = X^*_m = 1$, и

$$\overline{S}_m = \frac{1}{m}(l + (m-l)MX_1),$$

то

$$MT_{par} = \frac{2m-1}{m}(l + (m-l)MX_1) -$$

$$-\frac{2}{m}\left\{\sum_{k=m-l+1}^{m}(k-1) + \sum_{k=1}^{m-l}(k-1)MX^*_k\right\}.$$

Для доказательства достаточно завершить начатые алгебраические преобразования.

Замечание 1. Случай $l = 0$ в теореме характерен для естественного протекания процесса обработки. Случай $l > 0$ связан с различными нарушениями в процессе подготовки задач и надежностью вычислительных средств.

Замечание 2. Пусть в условиях теоремы $F(x) = x$. Тогда

$$MX_1 = \frac{1}{2}, \quad M\max(X_1, X_2) = \frac{2}{3},$$

$$\Delta = \frac{(m-1)(m+l)}{12m} + \frac{l(l-1)}{3m} \geq 0.$$

Это означает, что последовательный план совпадает с параллельным лишь при $m = 1$. В частности, при $l = 0$, $\Delta = (m-1)/12$, а при $l = m, \Delta = (m-1)/2$.

Для произвольной функции распределения $F(x)$ и $l = 0$, можно записать:

$$T_{seq} = \frac{m+1}{2} MX_1,$$

$$T_{par} = (2m-1)MX_1 - (m-1)M\max(X_1, X_2),$$

$$\Delta = (m-1)\left(\frac{3}{2}MX_1 - M\max(X_1, X_2)\right).$$

Последнее означает, что планы существенным образом зависят от такого параметра распределения $F(x)$ как $\frac{3}{2}MX_1 - M\max(X_1, X_2)$.

Следует обратить внимание, что полученные аналитические результаты существенным образом используют гипотезу о симметричной неопределенности, которая в реальности выполняется далеко не всегда, если вообще справедлива. В этой ситуации для более четкого понимания процесса целесообразно прибегнуть к методам имитационного моделирования.

§4. Обсуждение результатов вычислительного эксперимента

Для исследования модели, описанной в предыдущих параграфах, был проведены следующие вычислительные эксперименты. Набор заданий, удовлетворяющий определенным свойствам, генерировался случайным образом. Далее в первом случае все задания пакета выполнялись одно за другим, при этом каждой из задач последовательно отдавался весь имеющийся в наличии вычислительный ресурс. Во втором случае на выполнение был поставлен весь пакет одновременно, т.е. имеющиеся задания выполнялись параллельно. При этом мощность всех ИВУ первоначально была разделена на все задачи, а освободившиеся ресурсы (освободившиеся ИВУ) делились между оставшимися в зависимости от выбранной стратегии. Пакеты формировались как из задач со случайным временем окончания решения, равномерно распределенным на определенном временном интервале, так и из задач с фиксированным временем счета. Целью эксперимента являлось сравнение результатов обработки системой пакетов задач, обладающих различными свойствами, при параллельной и последовательной дисциплинах обслуживания.

Перейдем к более детальному описанию модели, использованной в вычислительном эксперименте. Производительность вычислительной системы принималась за единицу. В случае параллельного выполнения нескольких заданий считалось, что ресурс вычислительной системы может быть произвольным образом разделен между ними, и в любой момент времени существует возможность изменить распределение ресурса между задачами. Для последовательного выполнения предусматривалась возможность постановки на обработку нового задания в любой выбранный диспетчером момент времени. Предполагалось, что все операции — постановка задания на выполнение, завершение задания и перераспределение ресурсов производятся без потери времени.

Считалось, что в систему одновременно поступает пакет из m сгенерированных заданий, каждое из которых описывается случайной выборкой без возвращений. Предполагалось, что каждая такая задача имеет решение. Максимально возможное время решения каждой задачи $\overline{\tau_i}, i = \overline{1, m}$, для каждой группы экспериментов определялось отдельно, при этом длительность обработки каждого задания τ_i полагалась либо случайной величиной, равномерно распределенной на отрезке $[0; \overline{\tau_i}]$, либо была зафиксирована и равна $\overline{\tau_i}$. Кроме того, для каждого задания назначался директивный срок окончания его выполнения (дедлайн), $d_i, i = \overline{1, m}$.

Рассматривались четыре стратегии выполнения заданий:
I. Последовательное выполнение. В каждый момент времени системой выполняется ровно одно задание, при этом весь вычислительный ресурс отдается этому заданию. Когда завершается текущее задание, начинает выполняться следующее. Задания выполняются в случайном порядке.
II. Последовательное выполнение согласно стратегии SPT ($Shortest\ Processing-time\ Sequencing$). Следуя [2], в качестве набора априорных оценок длительностей работ используем величины $\overline{\tau_i}$ — максимально возможное время решения i-й задачи. Таким образом, имеющийся пакет задач был упорядочен по возрастанию величины $\overline{\tau_i}$ и перенумерован: на первом месте оказалось априорно самое "короткое" задание, на последнем — самое "объемное". Напомним, что реальное время обработки задания — случайная величина, поэтому такой порядок лишь наиболее вероятен после реализации.
III. Параллельное выполнение с предпочтением "срочных" относительно дедлайна заданий. Это означает, что все задания начинают выполняться одновременно, при этом ресурс между

заданиями распределяется пропорционально величине отношения $\overline{\tau_i}/d_i$. После завершения какого-либо задания освободившийся ресурс распределяется между оставшимися также пропорционально $\overline{\tau_i}/d_i$.
IV. Параллельное выполнение с предпочтением "коротких" (наименее "объемных") заданий. Это означает, что все задания начинают выполняться одновременно, при этом первоначально ресурс между заданиями распределяется пропорционально величине $\overline{\tau_i}/d_i$. После завершения какого-либо задания весь освободившийся ресурс отдается тому заданию, у которого меньше максимально возможное время счета.

Для сравнения параллельного и последовательного выполнения заданий был проведен ряд вычислительных экспериментов с целью выявления особенностей процедуры их прохождения. Для каждой модели генерировалось 10^6 случайных реализаций наборов данных, вычислялись выборочные средние и среднеквадратичные отклонения исследуемых величин. Результаты счета сравнивались по следующим показателям: среднее время завершения i-го задания, средняя разность моментов завершения i-го и i–1-го заданий, их среднеквадратичное отклонение и др. Наиболее интересные факты, обнаруженные в ходе экспериментов, приводятся ниже в виде графиков и таблиц.

Первый эксперимент.

В первом эксперименте рассматривались две стратегии — последовательное выполнение заданий и параллельное с равномерным распределением освобождающихся ресурсов.

Каждый пакет содержал 12 заданий, время завершения выполнения каждого задания являлось случайной величиной, равномерно распределенной на отрезке от 0 до 14/12 часа, при этом максимально возможное время выполнения было равно 14/12 часа.

Результаты эксперимента соответствуют основным выводам §3, а именно — среднее время пребывания в системе при последовательном выполнении заданий оказывается меньше, чем при параллельном, что во многом подтверждает адекватность выбранной математической модели. При этом если при последовательном выполнении задания покидают систему в среднем через одинаковые интервалы, то при параллельной обработке моменты окончания сдвинуты к концу временного интервала, а именно — задания покидают систему через временные отрезки, величины которых сокращаются с постоянной скоростью. Заметим, что вне зависимости от использованной стратегии последнее задание завершается в один и то же момент времени.

По мере завершения заданий им присваивались порядковые номера. На рис. 1 помещен график зависимости момента выхода заданий из системы от присвоенного порядкового номера задания, на рис. 2 — график зависимости длительности интервалов между моментами выхода заданий из системы от порядкового номера задания. Линии с незакрашенными треугольниками иллюстрируют последовательное выполнение, и с незакрашенными квадратами — параллельное выполнение.

Вторая группа экспериментов.

Во второй группе также рассматривались две стратегии — последовательное выполнение заданий и параллельное с равномерным распределением освобождающихся ресурсов.

Каждый пакет содержал 12 заданий со случайным временем выполнения. Дедлайн был взят одинаковым для всех заданий и достаточно большим, чтобы успеть выполнить все задания. При этом рассматривались два варианта распределения времени выполнения заданий.

В первом варианте время завершения выполнения десяти заданий являлось случайной величиной, равномерно распределенной на отрезке от 0 до 2 часов, при этом максимально

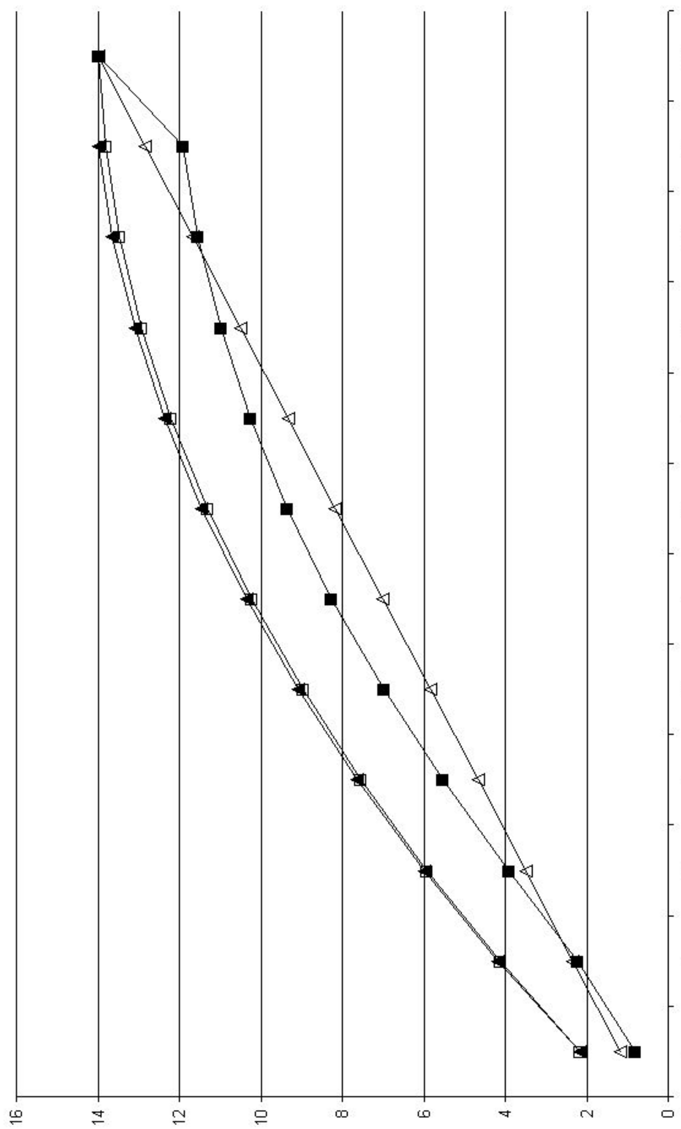

Рис. 1

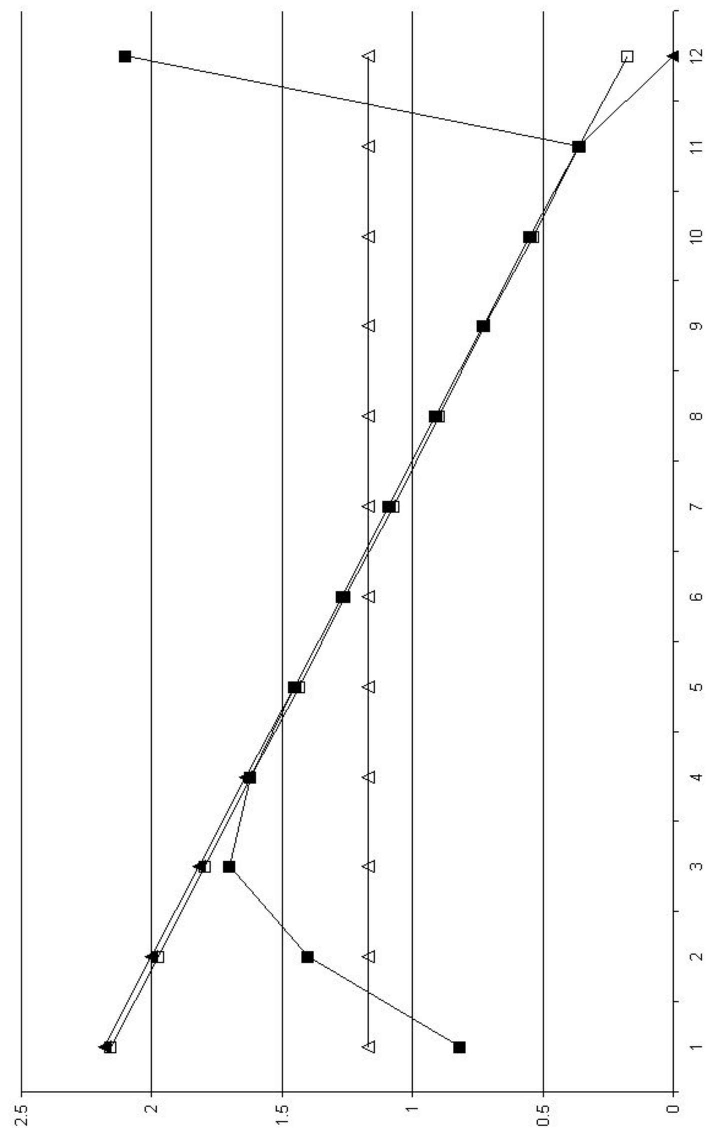

Рис. 2

возможное время выполнения было равно двум часам; два оставшихся задания имели детерминированное время выполнения — 2 часа.

Результаты этого эксперимента также помещены на рис. 1, 2. Линии с незакрашенными треугольниками иллюстрируют последовательное выполнение, с закрашенными треугольниками — параллельное выполнение. Заметим, что линия, иллюстрирующая этот эксперимент при параллельной обработке практически совпадает с линией, описывающей предыдущий эксперимент (также параллельная обработка), а линии, описывающие последовательное выполнение совпадают точно. Также обратим внимание на то, что в случае применения параллельной дисциплины выполнения временные расстояния между моментами завершения задач при параллельной обработке, как и для предыдущей модели, сокращаются, но две последние задачи выходят из системы одновременно.

Во втором варианте эксперимента время завершения выполнения десяти заданий являлось случайной величиной, равномерно распределенной на отрезке от 0 до 2 часов, при этом максимально возможное время выполнения было равно двум часам; два задания имели детерминированное время выполнения — 5 минут и 3 часа 55 минут. Данное предположение отвечает ситуациям реальной жизни, когда более длительный интервал в обработке задания может возникнуть, например, в результате сбоя в системе.

Результаты работы этого эксперимента на рис. 1, 2 иллюстрируют линии с незакрашенными треугольниками — последовательное выполнение, и с закрашенными квадратами — параллельное выполнение. Обратим внимание на следующие эффекты:

при параллельной обработке заданий выход из системы двух первых и двух предпоследних задач в среднем происходит раньше, чем при последовательной;

наличие задачи с фиксированной длительностью обработки в 5 минут для параллельной дисциплины дает стабильное время окончания первой задачи — ровно через 1 час после начала счета, и соответствующую фиксированную точку на графике. Размещение этой же задачи на случайном место при последовательной обработке фактически никак не влияет на среднее время окончания ни одной из задач (по-порядку) — на графике все точки совпадают с точками графиков, построенных для предыдущих моделей;

временные расстояния между "выходами" задач при параллельной обработке, как и для предыдущей модели, сокращаются, но только до тех пор, пока время завершения выполнения заданий — случайная величина, равномерно распределенная на отрезке от 0 до 2 часов. Как только это условие нарушается (для последней задачи время выполнения фиксировано, и составляет более двух часов), то соответствующий временной интервал резко увеличивается (рис. 2). На этом графике можно увидеть, что временной интервал между выходом одиннадцатой и двенадцатой задач больше соответствующего интервала между десятой и одиннадцатой. Эта зависимость, на наш взгляд, могла бы быть использована диспетчером для оценки процесса счета, выполняемого системой, с целью дальнейшей корректировки ресурса, выделенного на выполнение задания.

Обратим также внимание на поведение линий, описывающих зависимость длительностей интервалов между моментами выхода заданий из системы от номера задания. Для последовательной обработки эти линии точно совпадают во всех трех экспериментах. Для параллельной обработки линия для последнего эксперимента резко отличается от линий, полученных в предыдущих случаях (эксперимент 1, первый вариант эксперимента 2). Интервалы между моментами выхода задач

1-3 — увеличиваются, затем начинают убывать (линия для заданий 4-11 практически совпадает с линиями, описывающими предыдущие эксперименты), а затем мы наблюдаем резкий скачек в длительности. Это связано с тем, что самая "короткая" задача (время счета — 5 минут на всем ресурсе) выполняется фиксированное время, но быстрее, чем каждая первая задача в любом из предыдущих экспериментов. Таким образом, 1/12 ресурса в последнем эксперименте освобождается раньше, чем в стальных случаях.

По условию освободившийся ресурс равномерно добавляется одиннадцати незавершенным заданиям, поэтому вторая по счету задача (являющаяся минимальной по объему среди 10-ти заданий, имеющих максимально возможное время счета равномерно распределенное на отрезке от 0 до 2 часов) выходит из системы раньше, чем в предыдущих экспериментах, использующих параллельную стратегию (рис.1). Однако временной интервал между выходами первого и второго заданий при этом увеличивается вследствие того, что следующие 10 заданий имеют параметры, отличные от параметров первой задачи. Здесь использование параллельной стратегии как бы "выравнивает" ситуацию, приближая ее к результатам последних двух экспериментов, использующих параллельную стратегию. Аналогичная ситуация происходит с интервалом между выходами 11 и 12 задач, это также связано с тем, что последняя задача имеет параметры, отличные от 10 предыдущих задач.

Ситуация с третьей задачей иная. Действительно, вторая и третья задачи в последнем эксперименте выходят из системы через тот же промежуток времени, что и, соответственно, первая и вторая в двух предыдущих экспериментах (параллельная стратегия, рис.1). Однако временной интервал между выходами второй и третьей задач уже меньше соответствующего интервала между первой и второй, и далее эти интервалы продолжают убывать (рис. 2).

Третья группа экспериментов.

В третьей группе рассматривалось четыре стратегии выполнения заданий:
1). последовательное исполнение;
2). последовательное выполнение с применением упорядочивания SPT ($Shortest\ Processing - time\ Sequencing$);
3). параллельное выполнение с предпочтением "срочных" относительно дедлайна заданий.
4). параллельное выполнение с предпочтением "коротких" (наименее "объемных") заданий.

Далее на рис. 3-14 линия с незакрашенными треугольниками отвечает первой стратегии, линия с незакрашенными квадратами — второй стратегии, линия с закрашенными треугольниками — третьей стратегии, линия с закрашенными квадратами — четвертой стратегии. На рис. 3, 5, 7, 9, 11, 13 помещены графики зависимости момента выхода заданий из системы от порядкового номера задания, на рис. 4, 6, 8, 10, 12, 14 — графики зависимости интервалов между выходами заданий из системы от порядкового номера задания.

Было поставлено 6 подгрупп экспериментов. Каждый пакет содержал 12 заданий со случайным временем выполнения.

В первых трех подгруппах экспериментов дедлайн был одинаковым — 12 часов. Величины $\overline{\tau_i}, i = \overline{1, n}$ формировались так: случайным образом было выбрано 12 величин, равномерно распределенных от 0 до 1, затем они были нормированы так, чтобы сумма равнялась двенадцати. Полученные значения определили $\overline{\tau_i}, i = \overline{1, n}$.

В качестве $\tau_i, i = \overline{1, n}$ в первом случае (эксперимент 3-I-1) были выбраны 12 случайных величин, равномерно распределенных на соответствующих отрезках $[0, \overline{\tau_i}]$, во втором случае (эксперимент 3-I-2) — такое распределение имели только

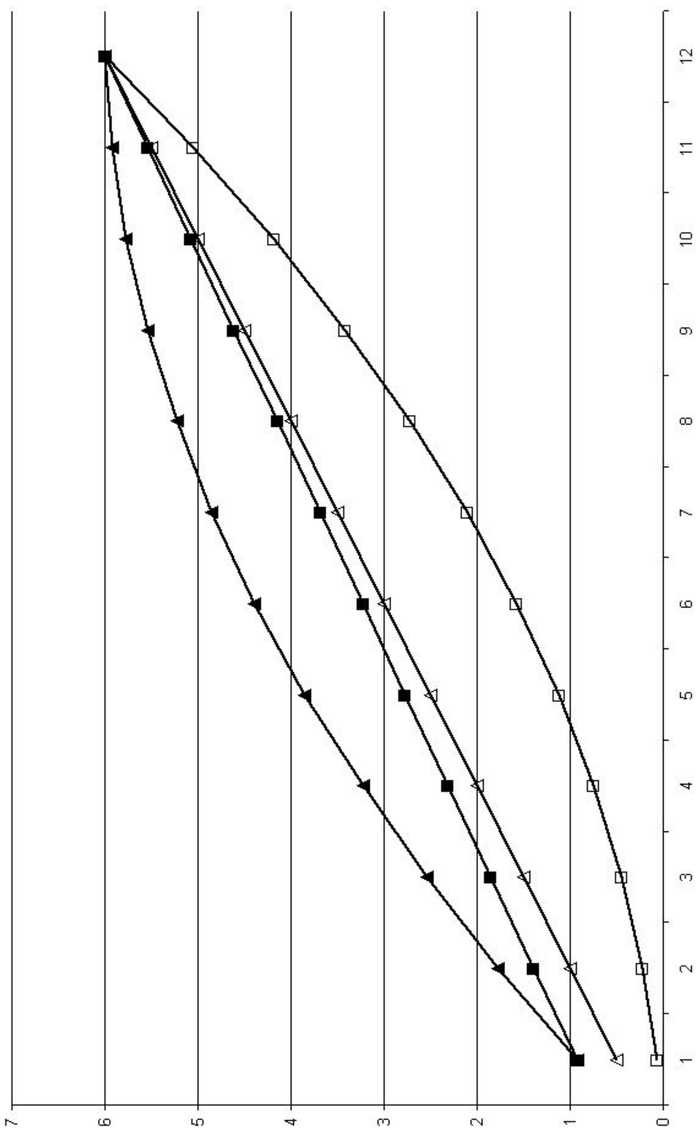

Рис. 3

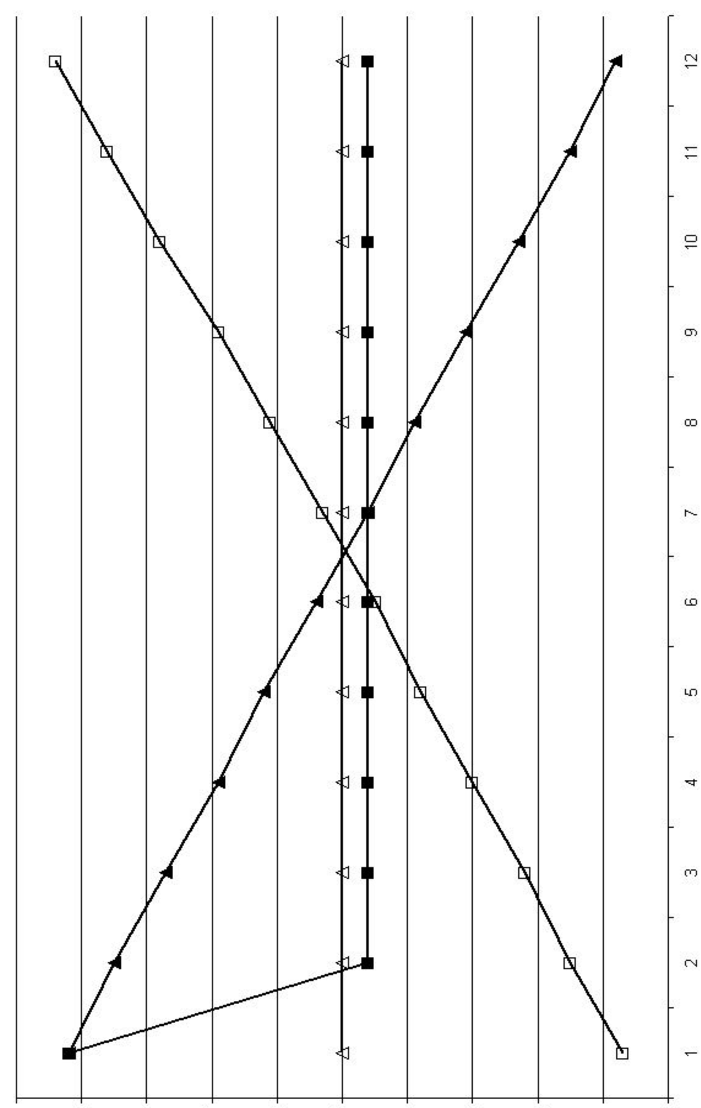

Рис. 4

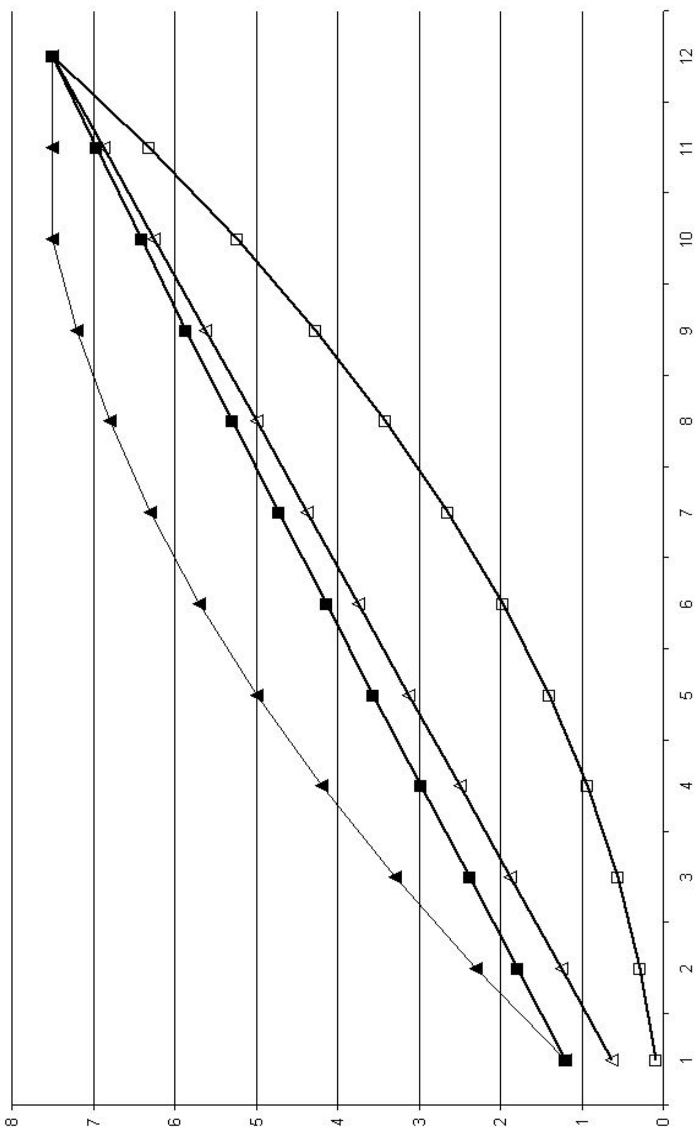

Рис. 5

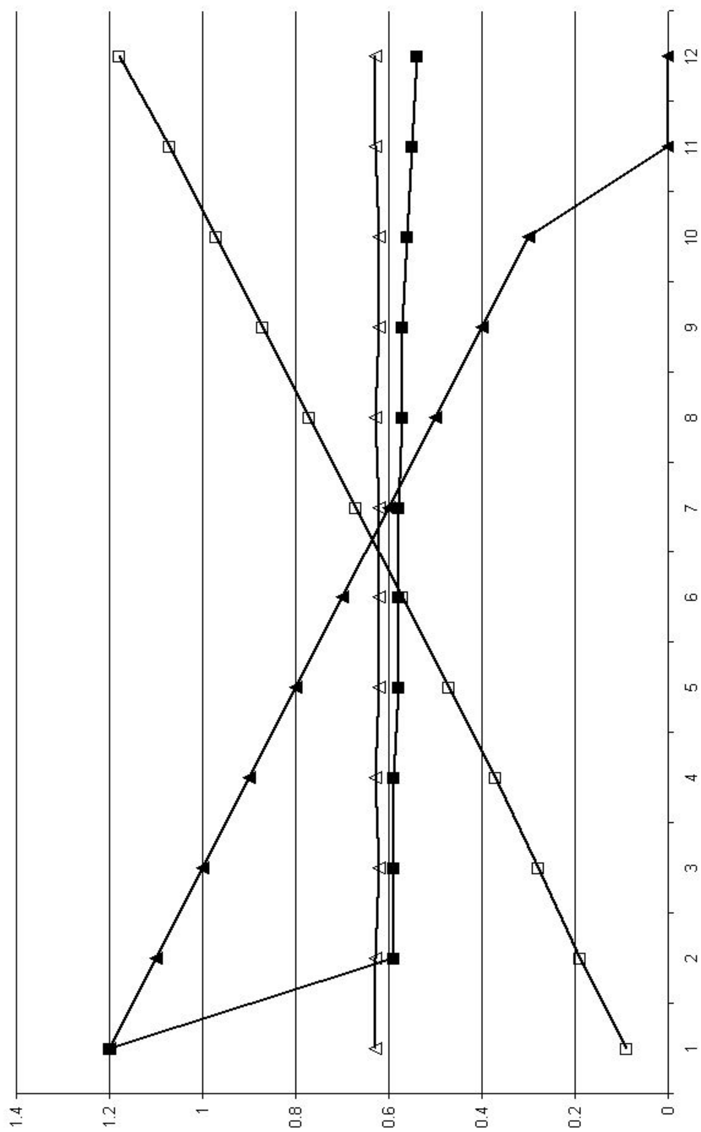

Рис. 6

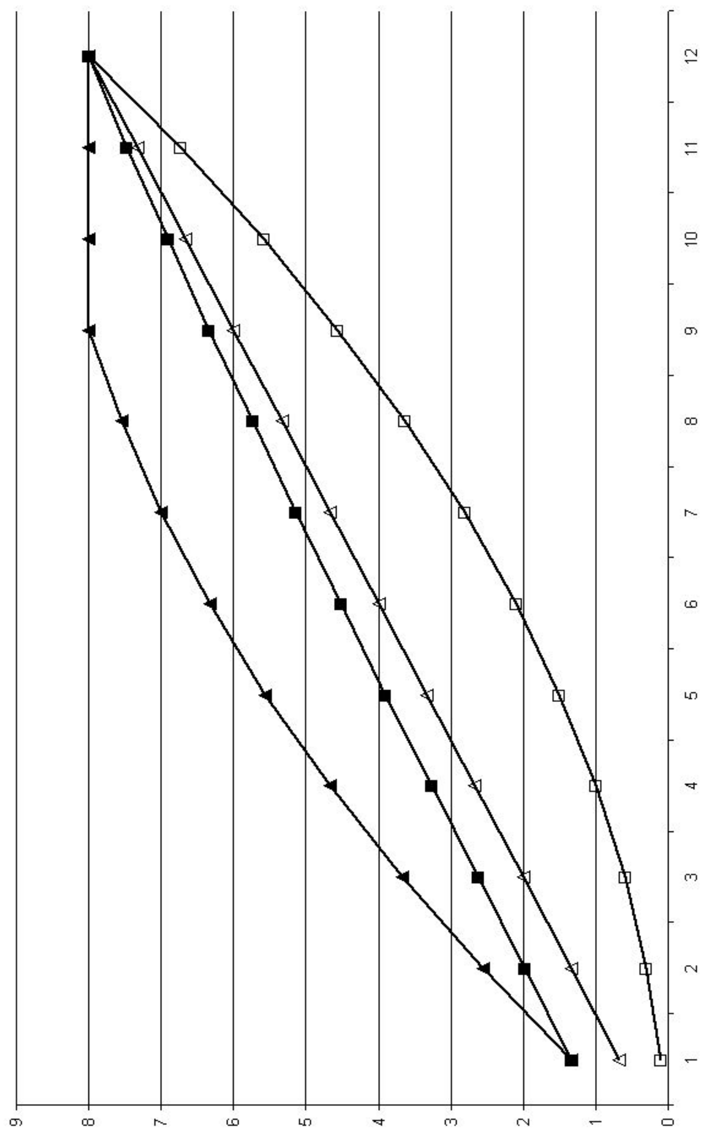

Рис. 7

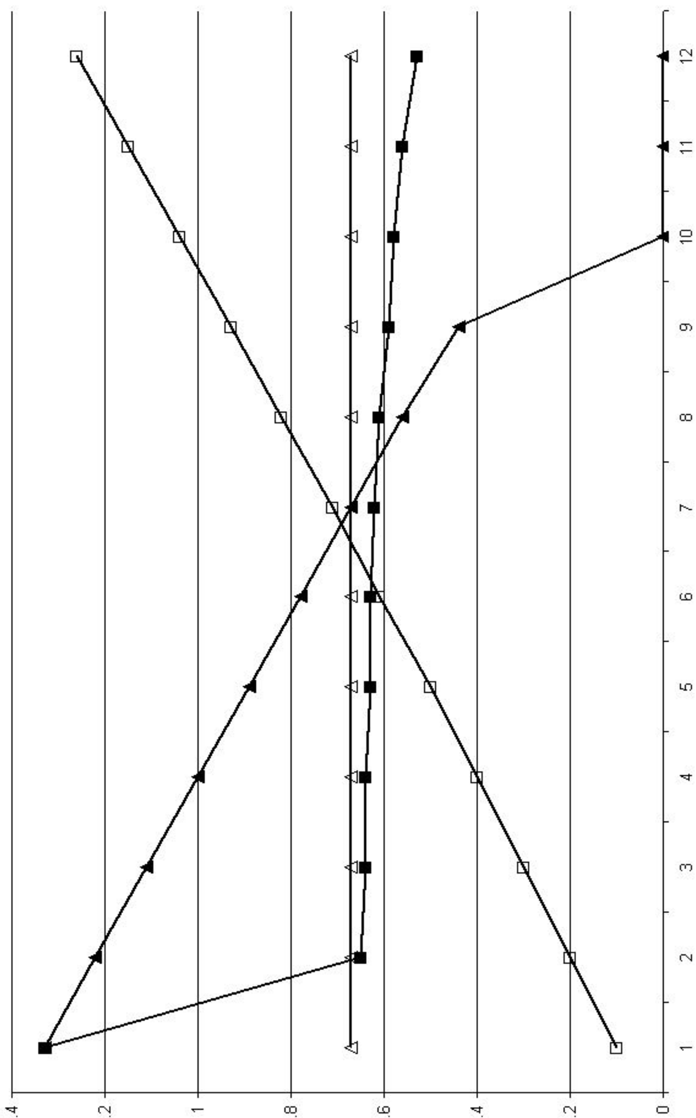

Рис. 8

9 задач, остальные выполнялись фиксированное время, равное максимально возможному, т.е $\overline{\tau_i}$. При этом номера заданий, которые выполняются до конца определялось случайным образом. Пакеты заданий для третьей подгруппы (эксперимент 3-I-3) были составлены аналогично с той лишь разницей, что соотношение задач, имевших случайное время решения, к задачам с фиксированным временем, было восемь к четырем. Заметим, что при таком формировании исходных данных для пакетов все задачи имеют не только одинаковый дедлайн, но и примерно равное среднее время счета.

Результаты соответствующих экспериментов приведены на графиках с рис. 3-8. По ним можно сделать следующие выводы:
1). Среднее время пребывания в системе одного задания для таким образом сформированных пакетов при последовательной дисциплине выполнения оказывается меньше, чем для параллельной, причем применение стратегии SPT предпочтительнее случайного упорядочивания (табл. 1.). Таким образом, оценка реальной длительности выполнения i-го задания по величине $\overline{\tau_i}$, сделанная нами в начале текущего параграфа, оказалась достаточно достоверной;

Табл. 1

Стратегии	1	2	3	4
Эксперимент 3-I-1	3,25	2,31	4,17	3,46
Эксперимент 3-I-2	4,06	2,88	5,38	4,40
Эксперимент 3-I-3	4,33	3,08	5,89	4,77

2). Стратегия передачи освободившегося ресурса единственной задаче, имеющей минимальное значение $\overline{\tau_i}$ (максимально возможное время решения), оказывается ближе по результатам к последовательной дисциплине со случайным порядком выполнения заданий, чем стратегия, использующая равномерное перераспределение (рис. 4, 6, 8). Применение стратегии передачи всего ресурса одной задаче нивелирует эффект одновременного выхода задач, возникающий при равномерном перераспределении;

3). На графиках 4, 6, 8 хорошо видно, что время между завершениями заданий при использовании последовательных стратегий не убывает, а при использовании параллельных стратегий — строго не возрастает;

4). Поскольку в экспериментах 3-I-2, 3-I-3 появляются задачи с фиксированным временем счета, равным максимально возможному, то кривые, описывающие последовательную обработку со случайным порядком заданий, на графиках 6, 8 располагаются выше, чем соответствующая кривая на графике 4. Подобные изменения происходят со всеми кривыми. Это иллюстрирует увеличение среднего времени пребывания в системе всех заданий;

5). Форма кривых для последовательной обработки фактически не меняется при переходе от эксперимента к эксперименту, а кривые, описывающие параллельную обработку с появлением заданий с фиксированным временем счета, приобретают более сложную форму. Так, например, линия с закрашенными треугольниками на рис. 4 не имеет изломов, а на рис 6, 8 они появляются; у линии с закрашенными квадратами на рис. 4 — один излом, далее она идет параллельно линии с незакрашенными треугольниками, т.е. фактически повторяет стратегию последовательной обработки со случайным порядком заданий, а на рис. 6, 8 у нее начинает "провисать хвост". Это говорит о том, что передача освободившегося ресурса более "короткому" заданию приводит к сокращению длительностей интервалов между выходами заданий из системы.

Рассмотрим следующие три подгруппы экспериментов. Для них исходные данные формировались следующим образом.

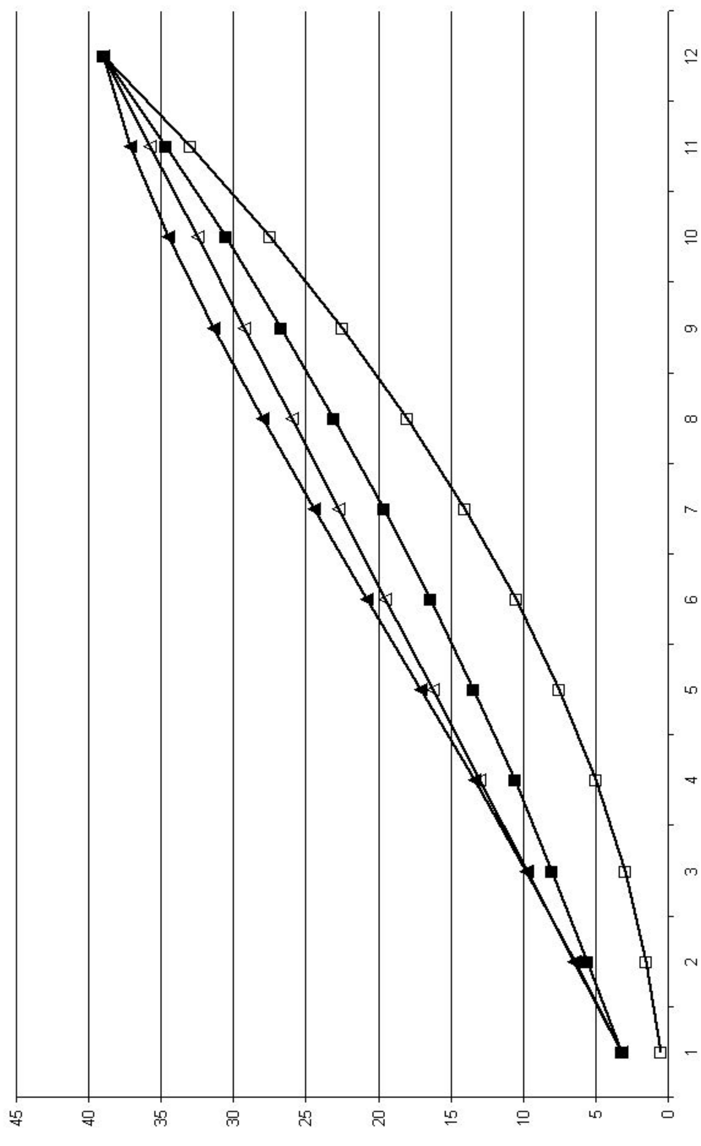

Рис. 9

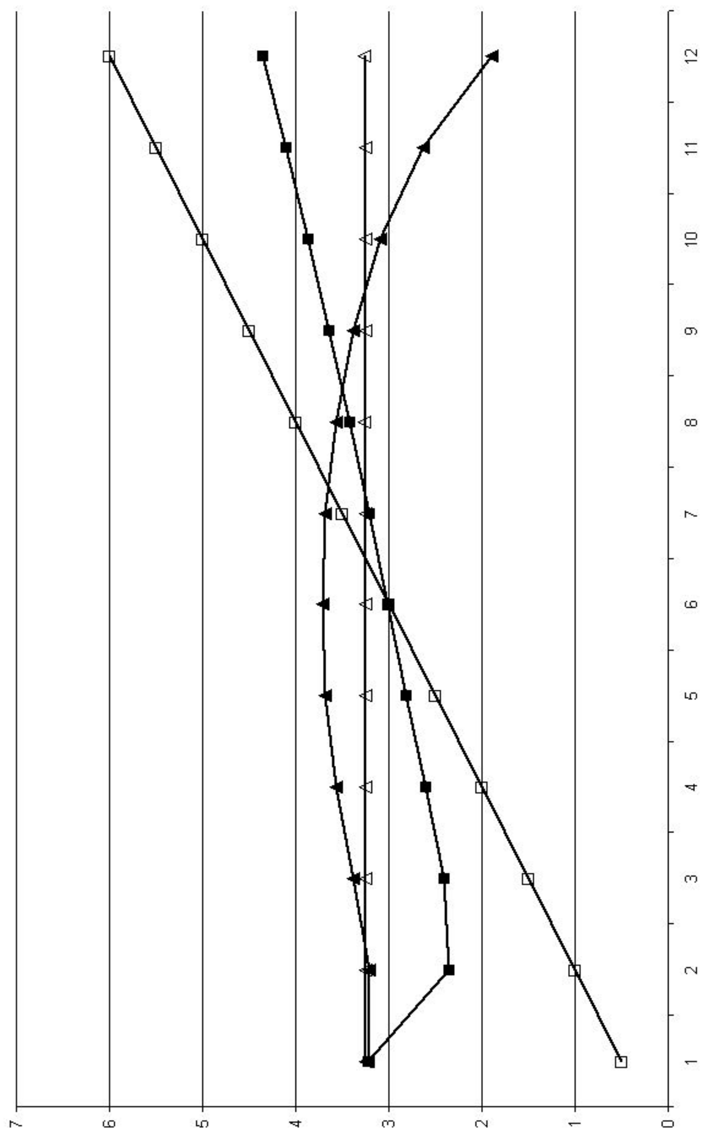

Рис. 10

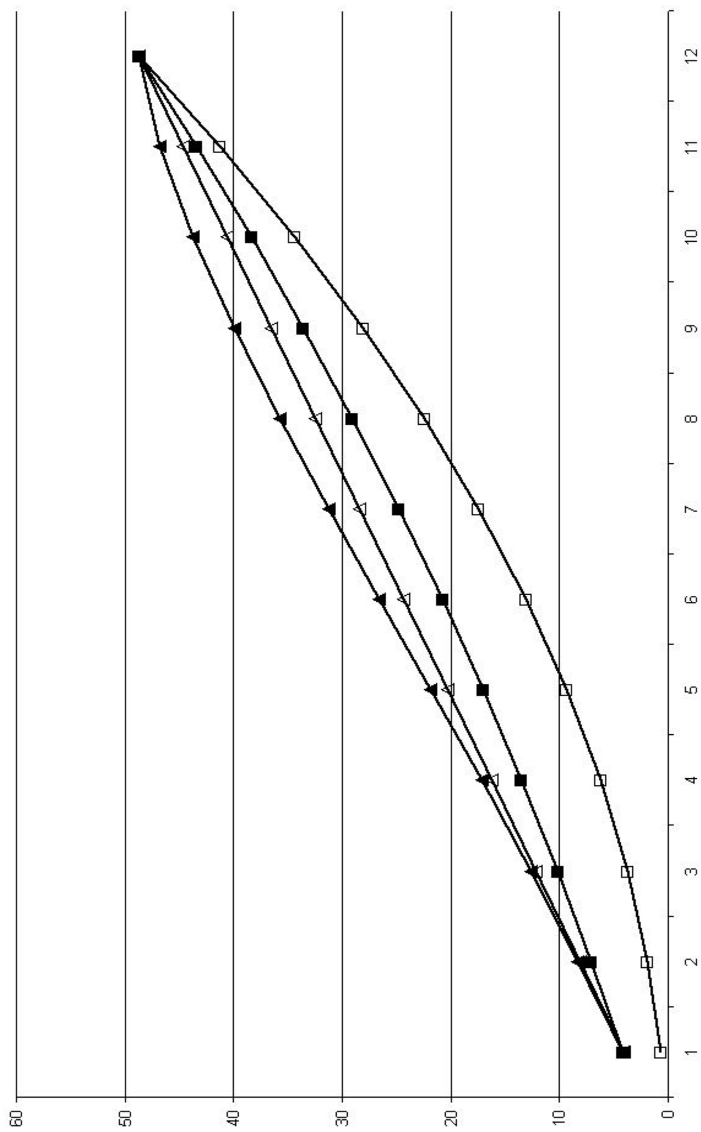

Рис. 11

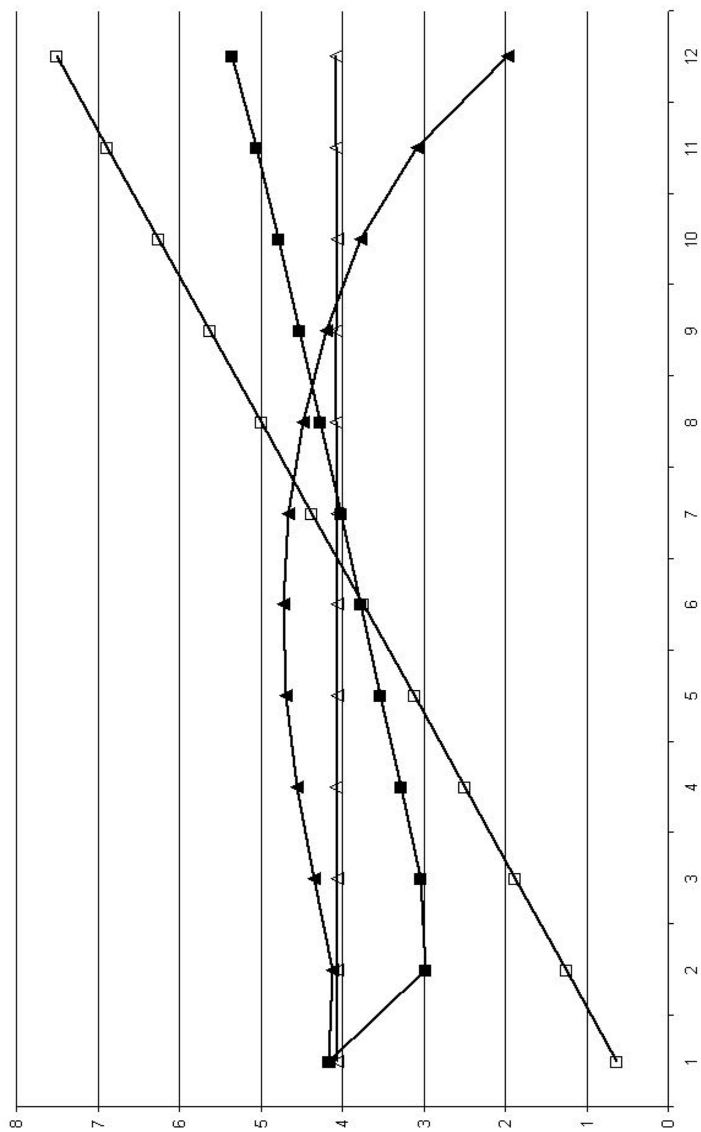

Рис. 12

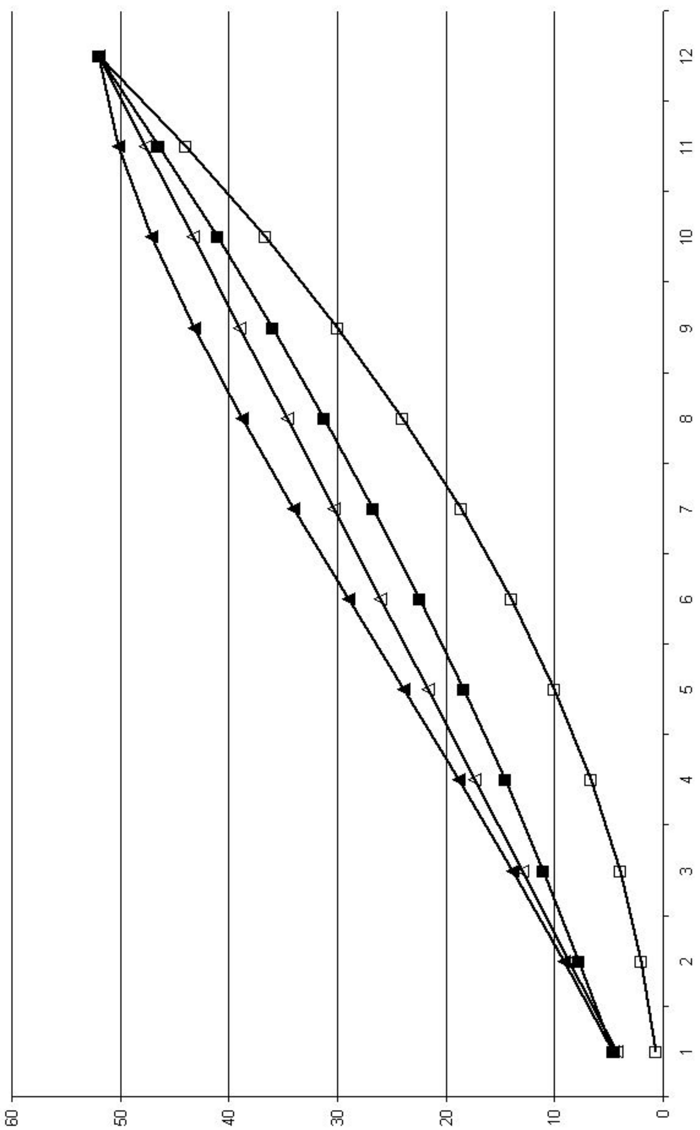

Рис. 13

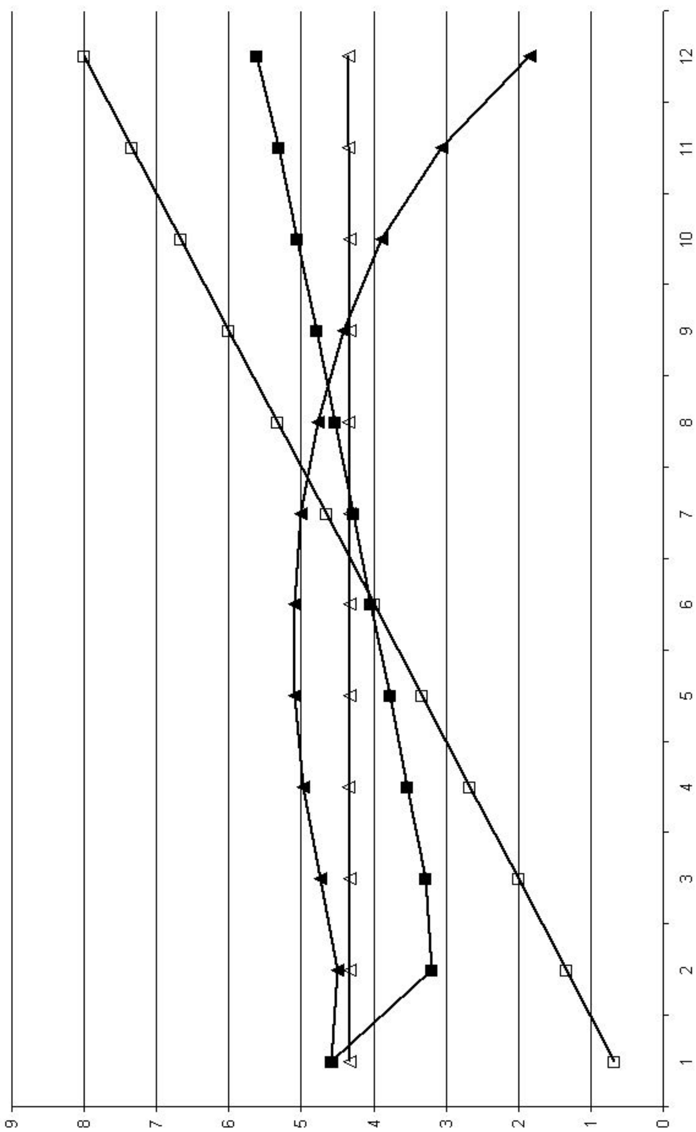

Рис. 14

Дедлайн для каждой i-й задачи всех трех подгрупп был равен $10i$, т.е. $d_i = 10i$. Каждая из $\overline{\tau_i}, i = \overline{1,n}$ была выбрана случайно из отрезка $[0, i]$, величины $\tau_i, i = \overline{1,n}$ были выбраны так же, как и в предыдущем эксперименте, т.е. в качестве $\tau_i, i = \overline{1,n}$ в первом случае были выбраны 12 случайных величин, равномерно распределенных на соответствующих отрезках $[0, \overline{\tau_i}]$, во втором случае — такое распределение имели только 9 задач, остальные выполнялись фиксированное время, равное максимально возможному, т.е $\overline{\tau_i}$. При этом номера заданий, которые выполняются до конца определялись случайным образом. Пакеты заданий для третьей модели были составлены аналогично с той лишь разницей, что соотношение задач, имевших случайное время решения, к задачам с фиксированным временем, было восемь к четырем.

Результаты соответствующих экспериментов приведены на графиках с рис. 9-14. По ним можно сделать следующие выводы:

1). Среднее время пребывания в системе одного задания для таким образом сформированных пакетов при последовательной дисциплине с применением стратегии SPT оказывается меньше, чем для остальных стратегий (табл. 2.);

Табл. 2

Стратегии	1	2	3	4
Эксперимент 3-II-1	21,13	15,17	22,11	19,25
Эксперимент 3-II-2	26,41	18,96	28,09	24,23
Эксперимент 3-II-3	28,17	20,22	30,38	26,03

2). Среднее время пребывания в системе при параллельной обработке заданий с передачей всего освободившегося ресурса задаче с наименьшим $\overline{\tau_i}$ оказывается меньше, чем при использовании стратегий 1 и 3. Это фактически означает, что существуют наборы задач и способы распределения освободившего-

ся ресурса, при котором среднее время пребывания в системе при параллельной обработке оказывается меньше, чем при последовательной со случайным порядком исполнения заданий;
3). Эффект, описанный выше в пункте 4 для предыдущей группы экспериментов, сохраняется. Здесь мы также можем наблюдать увеличение среднего времени пребывания в системе всех заданий;
4). В случае, когда исходные данные для заданий в пакете отличаются по величине (а именно — каждая из $\overline{\tau_i}, i = \overline{1, 12}$ выбиралась случайно из отрезка $[0, i]$), то прямые, характеризующие время между окончаниями выполнения заданий для последовательных дисциплин остаются такими, как будто различий в данных нет. В случае применения параллельных дисциплин ломаные одного вида трансформируются в ломаные другого вида. Для того, чтобы в этом убедиться достаточно сравнить графики с рис. 4, 6, 8 и 10, 12, 14. Например, для схожих экспериментов 3-I-2 и 3-II-2 при применении дисциплины параллельного выполнения заданий с передачей освободившихся ресурсов единственной задаче соответствующая ломаная (с закрашенными квадратами) на графике 6 сначала убывает резко, а затем постепенно, а на графике 12 линия, описывающая тот же способ распределения ресурсов, сначала убывает, а затем начинает почти линейно возрастать.

Подводя итог данного параграфа, следует заметить, что при разработке диспетчерских правил могут быть учтены следующие особенности параллельной дисциплины обслуживания:
1). Первыми решаются "короткие" задачи, самые "длинные" — в конце, что позволяет диспетчеру прекращать выполнение последних, например, в силу предположения о некорректности условий постановки задания или сбое в работе ИВУ. В результате удается следовать простому правилу сортировки, обычно трудно реализуемому на практике — решать "короткие" зада-

чи, затем "длинные и, наконец, "снимать" со счета скорее всего "нерешаемые заведомо невыполнимые задачи;
2). В случаях, когда время решения задачи — случайная величина, имеющая равномерное распределение на заданном отрезке, интервалы между моментами завершения заданий при использовании параллельной стратегии уменьшаются. Бо́льшая часть заданий выполняется ближе к концу интервала обслуживания, когда через короткие промежутки времени систему покидает основная группа решенных задач. Учет последнего обстоятельства позволяет диспетчеру планировать завершение задач точно в заданные промежутки времени.

В результате возникает реальная возможность согласовывать график решения задач с директивными сроками их выполнения, а также по ходу счета обоснованно и жестко корректировать сам график.

Заключение

Статистические выводы и анализ результатов вычислительного эксперимента показали, что некоторые общепринятые рекомендации по управлению прохождением заданий со случайным временем исполнения оказываются неадекватными реальным условиям, поскольку речь здесь идет не о детерминированной неопределенности времени исполнения заданий, а об его априорной случайной неопределенности.

При рассмотрении систем прохождения заданий с убывающей функцией актуальности, даже в случае аддитивного функционала полезности выполненных работ, остаются нерешенные математические и вычислительные проблемы выбора оптимальных стратегий.

Существует потенциальная возможность выбора неаддитивных функций полезности, использование которых приводит к существенному повышению практической эффективности применения адаптивных параллельных планов прохождения заданий со случайным временем исполнения и убывающей функцией актуальности.

Литература

1. Танаев В.С., Гордон В.С., Шафранский Я.Н. Теория расписаний. Одностадийные системы. М.: Наука, 1984.

2. Конвей Р.В., Максвелл В.Л., Миллер Л.В. Теория расписаний. М.: Наука, 1975.

3. Голосов П. Е. Технология распределения типовых заданий со случайным временем выполнения и убывающей функцией потери ценности решения в распределенной разнородной вычислительной среде. // Научный сервис в сети Интернет: масштабируемость, параллельность, эффективность: Труды Всероссийской суперкомпьютерной конференции. М.: МГУ, 2009. С. 114-116.

4. Голосов П. Е. Планирование заданий с временной функцией потери ценности решения в сетевой среде распределенных вычислений: Дис. ... канд. техн. наук. Москва. 2010.

5. Дэйвид Г. Порядковые статистики. М.: Наука, 1979.

СОДЕРЖАНИЕ

Список литературы

[1] Танаев В.С., Гордон В.С., Шафранский Я.Н. Теория расписаний. Одностадийные системы. М.: Наука, 1984.

[2] Конвей Р.В., Максвелл В.Л., Миллер Л.В. Теория расписаний. М.: Наука, 1975.

[3] Голосов П. Е. Технология распределения типовых заданий со случайным временем выполнения и убывающей функцией потери ценности решения в распределенной разнородной вычислительной среде. // Научный сервис в сети Интернет: масштабируемость, параллельность, эффективность: Труды Всероссийской суперкомпьютерной конференции (21-26 сентября 2009 г., г. Новороссийск). М.: МГУ, 2009. С. 114-116.

[4] Голосов П. Е. Планирование заданий с временной функцией потери ценности решения в сетевой среде распределенных вычислений: Дис. ... канд. техн. наук. Москва. 2010.

[5] Дэйвид Г. Порядковые статистики. М.: Наука, 1979.