КВАЗИПЛАНИРОВЩИК ДЛЯ ИСПОЛЬЗОВАНИЯ ПРОСТАИВАЮЩИХ ВЫЧИСЛИТЕЛЬНЫХ МОДУЛЕЙ МНОГОПРОЦЕССОРНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ ПОД УПРАВЛЕНИЕМ СУППЗ

А.В. Баранов, Е.А. Киселёв, Д.С. Ляховец

Межведомственный суперкомпьютерный центр РАН

Системы пакетной обработки многопроцессорных вычислительных систем

Многопроцессорная вычислительная система (ВС), как правило, состоит из управляющей ЭВМ и вычислителя (см. рис. 1). Вычислитель представляет собой совокупность вычислительных модулей (ВМ), объединённых коммуникационной средой.

Пользователи отправляют задания на выполнение, формируя входной поток заданий. Каждое отправленное задание должно быть снабжено файлом специального вида – паспортом, содержащим информацию о требуемом количестве ВМ для счёта, планируемом времени выполнения и иных требованиях к ресурсам вычислителя. Поток заданий поступает на управляющую ЭВМ, где формируется очередь заданий. Дождавшееся своей очереди задание поступает на выполнение на указанном пользователем количестве ВМ. Вышеперечисленные функции (приём входного потока пользовательских заданий, ведение очереди заданий и запуск заданий на счёт) выполняет специальное программное обеспечение – система пакетной обработки (СПО), которая может быть размещена как только на управляющей ЭВМ, так и на управляющей ЭВМ и ВМ.

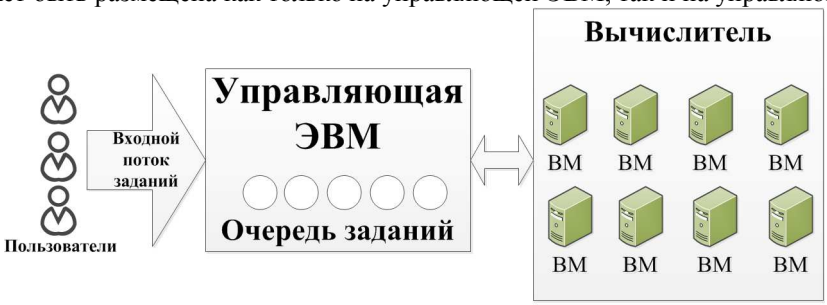


Рис. 1. Схема многопроцессорной вычислительной системы

Алгоритм обратного заполнения планировщика СПО СУППЗ

В отечественных многопроцессорных вычислительных системах много лет используется Система управления прохождением параллельных заданий (СУППЗ) – система планирования запуска и прохождения параллельных заданий, созданная в Институте прикладной математики им. М.В. Келдыша Российской академии наук (ИПМ им. Келдыша РАН) и Межведомственном суперкомпьютерном центре (МСЦ) РАН в 2001 году [1].

СУППЗ ведёт очередь параллельных заданий, обеспечивая по возможности полную загрузку вычислителя. В СУППЗ планирование осуществляется на основе алгоритма обратного заполнения (т.н. backfill–планировщик). В СУППЗ алгоритм обратного заполнения был реализован впервые в мировой практике построения СПО для многопроцессорных ВС, на основе этого алгоритма С.В. Шарфом (ИММ УРО РАН г. Екатеринбург) был реализован планировщик – сервер очередей, являющийся ядром СУППЗ [2].

Алгоритм обратного заполнения позволяет использовать простаивающие процессоры для запуска заданий с низким приоритетом вне очереди, если их выполнение не повлияет на время старта более приоритетных заданий. Такое возможно, например, в случае, если для задания А, стоящего в очереди ранее, недостаточно ресурсов, и при этом задание Б, стоящее в очереди позже, успеет завершиться до момента, когда освободится достаточное количество ресурсов для запуска задания А. Никакое менее приоритетное задание не может занять процессоры так, чтобы это отодвинуло старт более приоритетного. Планировщик определяет время завершения выполняющихся заданий и освобождения занятых процессоров, используя заказанное время выполнения, указанное в паспорте задания.

Эту особенность алгоритма обратного заполнения можно использовать для повышения загрузки вычислителя. Для этого необходимо отслеживать простаивающие процессоры и помещать в очередь такие задания, которые поступят на выполнение без ожидания в очереди.

Показатели качества планирования

Обычно выделяют следующие показатели качества планирования:

1. Утилизация ресурсов (загрузка вычислителя).

2. Среднее значение времени нахождение задания в очереди относительно заказанного времени его счёта.

Под **утилизацией ресурсов** будем понимать отношение задействованного при выполнении пользовательских заданий параллельного ресурса (совокупности ВМ) ко всему объёму параллельного ресурса. Если в течение 5 часов работали 10 ВМ, и в течение каждого часа один процессор простаивал в ожидании задания (см. рис. 2), то утилизация составит 80%.

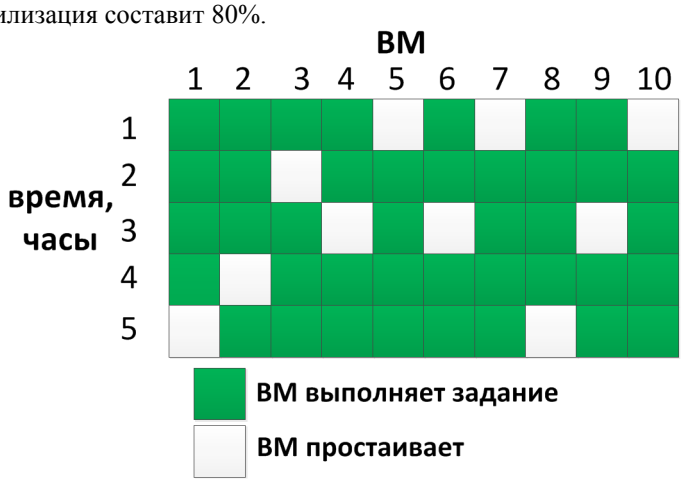


Рис. 2. Расчёт утилизации ресурсов вычислителя

Согласно статистическим данным работы [3] утилизация ресурсов систем под управлением СУППЗ находится на уровне 95%. Для больших ВС простаивание ресурсов около 5% оказывается весьма значительным, поскольку это эквивалентно тому, что один из 20 ВМ не работает. Для системы в 2000 ВМ в среднем простаивает 100 ВМ.

Предположим, что Q_i – время, прошедшее с момента попадания i-го задания в очередь до начала его выполнения, R_i – заказанное время выполнения i-го задания. Величину *времени нахождения задания в очереди относительно заказанного времени его счёта* обозначим как T_i:

$$T_i = \frac{Q_i}{R_i} \qquad (1)$$

Тогда для k заданий можно определить величину T_{mid} – **среднее значение времени нахождения задания в очереди относительно заказанного времени счёта**:

$$T_{mid} = \frac{\sum_{i=1}^{k} T_i}{k} = \frac{1}{k} * \sum_{i=1}^{k} \frac{Q_i}{R_i} \qquad (2)$$

Приведение ко времени счёта осуществляется потому, что для разных заданий одно и то же время ожидания может означать разное качество планирования. Например, время ожидания 30 минут будет вполне удовлетворительным для задания с заказанным временем счёта 10 часов, но то же время ожидания (30 мин) будет неприемлемым для задания со счётом в 5 минут. Очевидно, что чем меньше T_{mid}, тем выше качество планирования.

Повышение утилизации ресурсов за счет использования квазипланировщика

Современные высокопроизводительные ВС могут состоять из значительного (несколько сотен или тысяч) числа ВМ и обслуживать интенсивный входной поток пользовательских заданий. Некоторые ВМ будут неизбежно простаивать. Так, например, если следующему в очереди заданию В требуется 10 ВМ, а на текущий момент доступно лишь 9, то эти 9 ВМ будут простаивать, пока освободится ещё 1 ВМ, после чего задание В запустится на требуемых ему 10 ВМ (см. рис. 3).

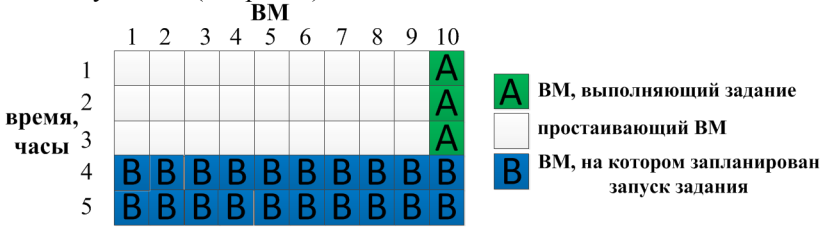


Рис. 3. Задание В ожидает запуска через 3 часа, 9 ВМ простаивают

Назовём совокупность простаивающих ВМ **окном** в плане запуска заданий. Под **размером окна** будем понимать количество простаивающих ВМ или процессоров и время их простоя. На рис. 1.3 приведено окно размером 9 ВМ на 3 часа (состоит из белых квадратов). Аналогично, **размером задания** назовём требуемое заданию число ВМ или процессоров и заказанное время счёта задания. Если размеры задания не превышают размера окна, то задание может быть запущено в окне плана запуска заданий (т.е. на простаивающих ВМ) без ожидания в очереди. Запуск такого задания будем называть **заполнением окна** плана запуска заданий.

Если в очереди найдётся задание С, размеры которого не превышают размеров окна, то алгоритм обратного заполнения позволит запустить задание С на счёт. Запуск задания С не задержит старт более приоритетного задания Б. Например, низкоприоритетное задание С размером 1 ВМ на 2 часа будет запущено на счёт в окне размером 9 ВМ на 3 часа без ожидания в очереди (см. рис. 4).

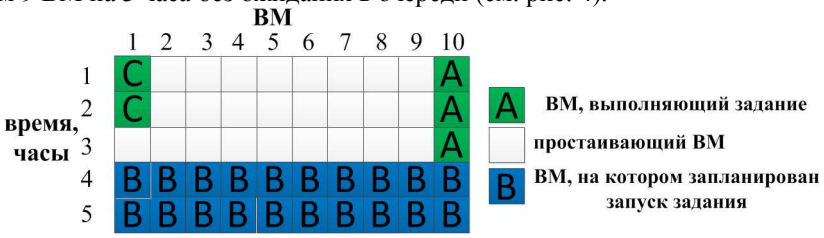


Рис. 4. Низкоприоритетное задание С запущено на 1 ВМ и 2 часа счёта

Авторами разработано программное средство (ПС) определения окон в плане запуска заданий планировщика СУППЗ, которое способно определять размеры окон текущего расписания. Размер окна представляет собой максимальный размер задания, которое будет запущено на счёт без ожидания в очереди.

Существует категория заданий, для которых требование к числу ВМ не является строгим. Такие задания предпочтительно запустить на 9 ВМ без ожидания, чем на 100 ВМ, но с длительным ожиданием. Одним из решений для таких пользователей может стать квазипланировщик, позволяющий указывать в паспорте нижнюю и верхнюю границы требуемого числа ВМ. Например, «9-100» должно означать, что задание предпочтительно запустить на 100 ВМ, однако возможно запустить и на любом количестве ВМ от 9 до 100, если это позволит уменьшить время пребывания в очереди. Квазипланировщик (см. рис. 5) определяет окна в плане запуска, принимает от пользователей задания и ставит в очередь планировщику СУППЗ задания, для которых достаточно простаивающих ресурсов. В силу особенности алгоритма обратного заполнения, СУППЗ будет запускать полученные от квазипланировщика задания на простаивающих ресурсах без ожидания в общей очереди.

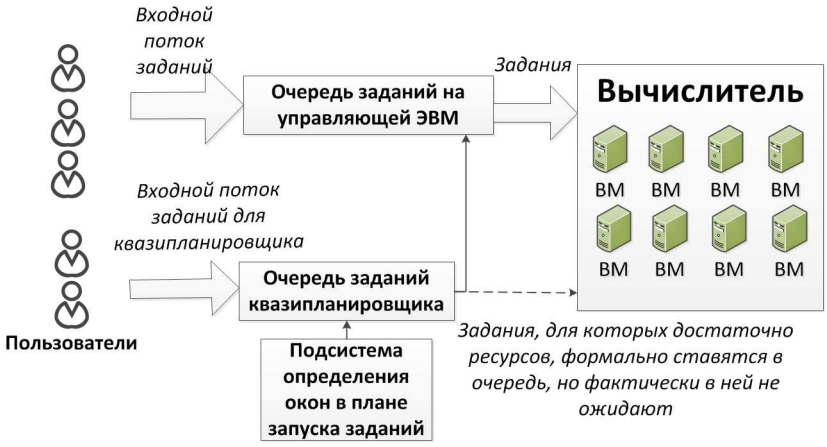
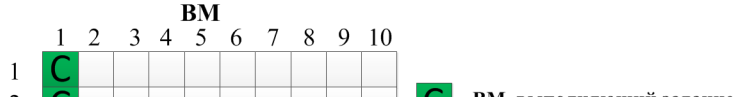


Рис. 5. Схема работы квазипланировщика

Термин «квазипланировщик» использован авторами по следующей причине. С точки зрения пользователя в системе появляется еще одна планируемая очередь для специального типа заданий. При этом реального планирования этой очереди не происходит, квазипланировщик лишь отслеживает окна в плане запуска общей очереди заданий и «подмешивает» в нее задания из своей очереди.

Попытки использования простаивающих ресурсов могут негативно отразиться на качестве обслуживания текущих заданий. В примере заполнения окна низкоприоритетным заданием (см. рис. 4) 1 ВМ из 9 свободных был задействован для расчёта задания С из конца очереди. Планирование запуска заданий происходит из заказанного пользователем времени счёта, а реальное время счёта может быть существенно меньше заказанного. Согласно статистике работы СУППЗ только у 10% заданий реальное время счёта совпадает с заказанным. Свыше 60% заданий завершаются за час и более до окончания заказанного времени счёта. Задание А может завершиться раньше, чем через 3 часа, и ожидаемый 1 ВМ освободится раньше планируемого.



Разница между заказанным и реальным временем счёта приводит к нежелательной ситуации (см. рис. 6) – менее приоритетное задание С, запущенное на 1 ВМ, мешает запуску более приоритетного В, и 9 ВМ будут ожидать завершения работы этого менее приоритетного задания в течение 2 часов. Подобные ситуации могут привести к тому, что качество обслуживания заданий в целом не улучшится, а ухудшится. Очевидно, что работа квазипланировщика будет оказывать влияние на показатели качества планирования СУППЗ, причем характер этого влияния заранее предсказать невозможно. Поэтому необходимо провести экспериментальное исследование влияния квазипланировщика на качество планирования.

Проведение подобных исследований на реальной ВС в реальном масштабе времени сопряжено с большими расходами вычислительных мощностей и времени, поскольку расчёт показателей качества эффективности планирования имеет смысл проводить за большие промежутки времени (больше нескольких дней). Это приводит к необходимости создания и использования симулятора СУППЗ, который позволит сократить время проведения исследования за счёт продвижения модельного времени.

Квазипланировщик одинаково функционирует на симуляторе и на реальной системе. Если использование квазипланировщика на симуляторе приведёт к положительным изменениям показателей эффективности, то можно будет задействовать квазипланировщик на реальной системе.

Свойства разработанного симулятора СУППЗ

Режим симуляции не предполагает реального выполнения заданий, поэтому на вход симулятора должен поступать автоматическим образом сформированный входной модельный поток заданий. Во время работы СУППЗ сохраняет параметры запущенных заданий в базе данных (БД) «Статистика» (см. рис. 7). В частности, в этой БД сохраняются время поступления задания в очередь, требуемое число процессоров и заказанное время счёта. Для создания входного модельного потока заданий из БД «Статистика» реальной СУППЗ можно извлечь параметры выполненных заданий.

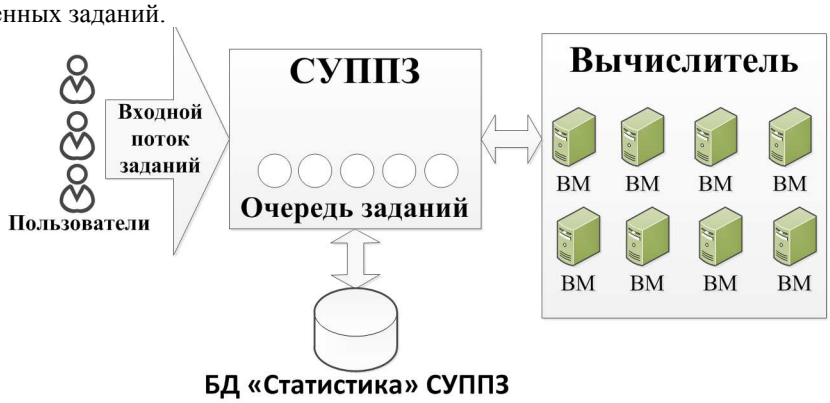


Рис. 7. Схема работы СУППЗ

Для исследования влияния работы квазипланировщика на показатели качества планирования СУППЗ необходимо запустить симулятор без квазипланировщика и с квазипланировщиком. Поиск оптимальных параметров квазипланировщика потребует многократного запуска симулятора с различными параметрами квазипланировщика.

Перед каждым запуском симулятора СУППЗ должна иметь одинаковое начальное состояние. Для получения начального состояния СУППЗ возможно сохранить файлы конфигурации, расписания, состояния планировщика и очереди (вместе с паспортами стоящих в очереди заданий) реальной СУППЗ.

Восстановив начальное состояние СУППЗ, необходимо изменить исследуемые настройки, влияющие на характеристики системы. К таким настройкам будем относить параметры квазипланировщика и множество входных заданий для него.

Свойства симулятора СУППЗ:

1. Симулятор функционирует без реального многопроцессорного вычислителя. Для этого задания запускают фиктивно, без реального выполнения.

2. Модельный поток заданий, поступающий на вход симулятора СУППЗ, имеет те же статистические характеристики, что и реальный поток заданий. Для создания модельного потока заданий создана и задействована специальная подсистема формирования модельного потока заданий. Эта подсистема извлекает из БД «Статистика» параметры выполненных в реальной СУППЗ заданий и представляет их в виде, пригодном для симулятора СУППЗ.

3. Запущенные на фиктивный счёт задания завершаются не по окончании заказанного времени счёта, а по окончании реального времени счёта из статистики работы реальной СУППЗ.

4. Для ускорения проведения симуляции возможно осуществлять продвижение модельного времени.

Схема работы квазипланировщика

Как уже отмечалось, существует категория заданий, для которых требование к числу ВМ не является строгим. Задания, выполнение которых возможно на произвольном числе процессоров из заданного диапазона,

образуют входной поток заданий для квазипланировщика (см. рис. 8). Из поступающих заданий квазипланировщик формирует собственную очередь.

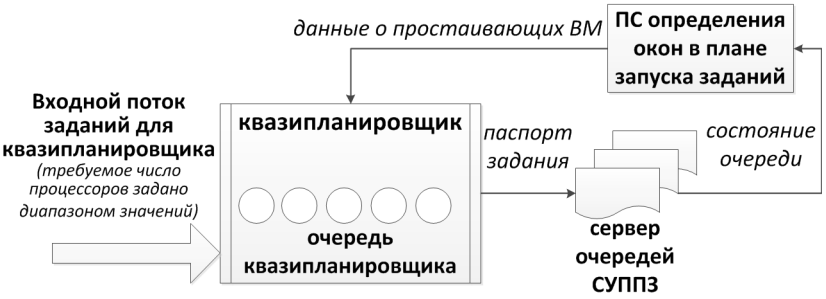


Рис. 8. Схема работы квазипланировщика

Подсистема определения окон в плане запуска заданий анализирует текущее состояние очереди и сообщает квазипланировщику размер ближайшего окна. Квазипланировщик рассматривает очередное задание в очереди и определяет возможность его запуска в ближайшем окне. Если запуск возможен, квазипланировщик формирует и ставит в очередь СУППЗ задание. Благодаря особенности алгоритма обратного заполнения формально поставленные в очередь СУППЗ задания фактически в ней не ожидают. Если запуск следующего задания невозможен, то квазипланировщик будет ожидать информацию о вновь появившихся окнах в плане запуска.

Если пользователь поставил в очередь квазипланировщика 10-минутное задание с требуемым числом процессоров в диапазоне от 9 до 100, и, согласно текущему плану запуска заданий, простаивает 50 процессоров в течение 15 минут, то возможно немедленно запустить на счёт задание на 50 процессоров на 10 минут.

Влияние квазипланировщика на показатели эффективности СУППЗ

Для проведения вычислительного эксперимента использовался журнал работы МВС-100К за неделю с 10:30 19.03.2014 по 10:00 26.03.2014. Начальное состояние СУППЗ было получено в 10:30 19.03.2014 во время штатной профилактики. Во время профилактики никаких пользовательских заданий на вычислителе не выполняется. В очереди на момент начала профилактики находилось 42 задания.

В журнале работы за недельный период были найдены все задания, которые при завершении не освободили занятые процессоры. Такие процессоры СУППЗ автоматически блокирует и исключает из состава вычислителя. Соответствующие задания в модельном потоке были снабжены дополнительным параметром – числом неосвобождённых при завершении процессоров.

За указанный период было запущено и завершено 4088 заданий согласно журналу работы реальной СУППЗ и 803 задания было запущено квазипланировщиком. Входной поток заданий для квазипланировщика состоял из однотипных заданий разных пользователей. Заказанное время счёта всех заданий 20 минут. Заказанное число процессоров указано диапазоном от 8 до 32. Реальное время счёта 20 минут.

Симуляция недели модельного времени заняла около трёх суток. Рассчитанные показатели утилизации для симулятора без квазипланировщика и с квазипланировщиком приведены в таблице 1.

Таблица 1. Расчёты утилизации

Дата	Утилизация без квазипланировщика	Утилизация с квазипланировщиком	Выгода
19.03.14	88,1	88,8	0,7
20.03.14	95,1	97,0	1,9
21.03.14	93,4	94,0	0,6
22.03.14	95,3	95,0	-0,3
23.03.14	93,1	94,9	1,8
24.03.14	96,3	96,9	0,6
25.03.14	91,9	93,0	1,1
Среднее арифметическое	93,31	94,23	

В большинстве случаев квазипланировщик оказывает положительное влияние на среднюю утилизацию ресурсов вычислителя. Отрицательное влияние имеет место быть из-за разницы между заказанным временем счёта и реальным временем счёта заданий основного входного потока СУППЗ.

Среднее значение времени нахождение задания в очереди относительно заказанного времени его счёта составило 0,62 для симулятора без квазипланировщика и 0,54 для симулятора с квазипланировщиком. При этом среднее время ожидания для заданий основного входного потока незначительно возросло и составило 0,63. Для

поставленных в очередь квазипланировщиком заданий среднее время ожидания в очереди находилось на уровне 0,001.

Выводы

Авторами впервые была поставлена и решена задача создания квазипланировщика для утилизации простаивающих ресурсов системы под управлением СУППЗ. В ходе вычислительного эксперимента на симуляторе СУППЗ пользователям квазипланировщика предоставлялась возможность без ожидания в очереди запускать на счёт задания, выполнение которых возможно на произвольном числе процессоров из заданного диапазона. Запуск таких заданий не оказал существенного отрицательного влияния на качество обслуживания текущих пользователей СУППЗ и позволил незначительно повысить среднюю утилизацию ресурсов.

Разработанный авторами квазипланировщик может быть внедрён в работу реальной СУППЗ на многопроцессорной вычислительной системе для повышения утилизации ресурсов и улучшения качества обслуживания пользователей квазипланировщика при отсутствии значительного влияния на среднее время ожидания в очереди заданий входного потока СУППЗ.

ЛИТЕРАТУРА:

1. Система управления прохождением параллельных заданий [Электронный ресурс]. URL: http://suppz.jscc.ru (дата обращения: 13.07.2014).
2. Шарф С.В. Обслуживание очереди задач и многофакторные приоритеты // Параллельные вычисления в ИММ УрО РАН [Электронный ресурс]. URL: http://parallel.imm.uran.ru/pubs/izhevsk03-scharf.htm (дата обращения: 13.07.2014).
3. А.В. Баранов, А.В. Киселев, В.В. Старичков, Р.П. Ионин, Д.С. Ляховец. Сравнение систем пакетной обработки с точки зрения организации промышленного счета // Научный сервис в сети Интернет: поиск новых решений: Труды Международной суперкомпьютерной конференции (17-22 сентября 2012 г., г. Новороссийск). М.: Изд-во МГУ, 2012. С. 506