

## Методика повышения эффективности управления ресурсоемкими задачами в распределенных вычислительных системах

*Аунг Чжо Мьо, А.А. Анисимов, Л.Г. Гагарина, Е.М. Портнов*

*Национальный исследовательский университет «Московский Институт  
Электронной Техники»*

**Аннотация:** В настоящее время достаточно остро стоит проблема повышения эффективности управления ресурсоемкими задачами в распределенных вычислительных системах (РВС). Анализ существующих алгоритмов диспетчеризации задач для традиционных вычислительных систем показал имеющиеся ограничения применения данных методов в гетерогенных системах, в результате чего могут происходить значительные задержки между переключением задач. Предложена функциональная математическая модель оценивания загруженности РВС, основанная на определении вероятности простоя обрабатывающего центра в качестве критерия оценивания загруженности системы. Использование данной модели обеспечит повышение эффективности управления ресурсоемкими задачами в распределенных вычислительных системах.

**Ключевые слова:** математическая модель, распределенная вычислительная система, обрабатывающий центр, загрузка.

В настоящее время широкое распространение получили распределенные вычислительные системы, в которых параллельно задействовано значительное количество обрабатывающих центров (ОЦ), позволяющие эффективно работать над ресурсоемкими задачами. В качестве примера такой задачи можно привести обработку результатов научных экспериментов, а также расчет и моделирование реальных физических процессов.

Одной из основных функций РВС является рациональное распределение ресурсов между пользователями и вычислительными задачами [1]. Для этих целей необходимы эффективные алгоритмы управления задачами, минимизирующие время ожидания и возможные потери информации. Выбор конкретного алгоритма управления определяется, как правило, классом задач, решаемых вычислительной системой и целями, которые необходимо достичь [2,3].

РВС может быть представлена в следующем виде (1):

$$W = \{W_{\text{выч}}, W_{\text{вз}}, W_{\text{пр}}, W_{\text{уп}}\}, \quad (1)$$

где  $W_{\text{выч}}$  – вычислительная подсистема РВС, представляющая собой совокупность обрабатывающих,  $W_{\text{вз}}$  – среда взаимодействия,  $W_{\text{пр}}$  – протоколы передачи данных между вычислительными узлами РВС в процессе обработки задач,  $W_{\text{уп}}$  – система управления.

Обрабатывающие центры РВС представляют множество  $P = \{P_1, \dots, P_n\}$  и могут представлять собой, например, процессор общего назначения, процессор специального назначения (например, цифровой сигнальный процессор и др.

Входной поток задач определяется как множество  $T = \{T_1, \dots, T_N\}$  задач, подлежащих исполнению на РВС с заданным набором ресурсов. Каждая задача из потока задач  $T_i \in T$  описывается выражением  $T_i = \{S_T, t, R\}$ , где  $S_T$  – источник задачи,  $t$  – момент времени поступления задачи в систему,  $R$  – ресурсоемкость задачи, определяющая совокупность требований задачи к ресурсам различного типа [4]. Задача статического управления задачами в РВС состоит в формировании очереди готовых к выполнению задач на основе плана распределения ресурсов. План распределения ресурсов представляет собой соответствие задач из очереди и вычислительных узлов, а также моментов времени предоставления требуемых ресурсов [5,6].

В качестве критериев эффективности плана распределения ресурсов обычно используются такие показатели эффективности, как: минимум среднего времени обработки задания; минимум количества требуемых процессоров; максимум загрузки вычислительных узлов; минимум суммарного времени выполнения всего набора задач; обработка задач за время меньше заданного; равномерное распределение вычислительных ресурсов.

Если РВС состоит из  $s$  обрабатывающих центров с пиковой производительностью  $p_1, \dots, p_s$  и работает с загруженностью  $z_1, \dots, z_s$ , то

реальную производительность  $\Psi$  системы можно представить следующим образом:

$$\Psi = \sum_{i=1}^s z_i p_i. \quad (2)$$

Из выражения (2) следует, что для достижения наибольшей реальной производительности РВС необходимо обеспечить наибольшую загруженность всех ее обрабатывающих центров. Исходя из этого, критерий эффективности управления ресурсоемкими задачами в РВС можно определить целевой функцией :

$$\Psi = \sum_{i=1}^s z_i p_i \rightarrow \max \quad (3)$$

Пусть  $\alpha_i(P_i)$  и  $\beta_i(P_i)$  – моменты начала и завершения выполнения задачи  $T_i \in T$  на обрабатывающем центре  $P_i$ . Зададим уровни наличия ресурсов, доступных обрабатывающему центру  $P_i \in P$  в момент времени  $t$ , следующим образом [7]:

$$r_i(P_i) = \{r_i^j(P_i)\}, i \leq j \leq J, t \in [0, T], \quad (4)$$

где  $J$  – число различных ресурсов. Верхнюю границу каждого ресурса, доступного вычислительному узлу  $P_i \in P$  обозначим  $r^j(P_i)$ , причем  $r_i^j(P_i) \leq r^j(P_i)$  для всех  $t$  и  $j$ .

Ограничения на составление расписаний устанавливаются исходя из имеющихся ресурсов и набора задач. Пусть  $T_t = \{T_i \in T \mid \alpha_i(P_i) \text{ и } \beta_i(P_i)\}$  – множество задач, выполняемых в момент времени  $t$ , тогда расписание является допустимым, если для всех  $T_i, T_j \in T$  выполняются условия :

$$\begin{cases} \beta_i(P_i) \leq T, T_i < T_j \Rightarrow \alpha_i(P_i) \leq \beta_i(P_i) \\ \sum_{T_i \in T} r_i^j(P_i) \leq r_j(P_i), t \in [0, T] \end{cases} \quad (5)$$

Длительность  $\tau_i(P_i)$  выполнения задачи  $T_i$  на обрабатывающем центре  $P_i$  рассчитывается во время работы алгоритма управления задачами в РВС исходя из значений, указанных пользователем на этапе постановки задачи в очередь на выполнение. При работе алгоритма управления задачами в РВС

необходимо масштабировать длительность выполнения процессов в соответствии с фактически используемым типом ОЦ. Новое значение величины  $\tau_i(P_i)$  определяется в соответствии с выражением:

$$\tau_i(P_i) = \tau_i(P_h) k^c(P_h, P_g) k^m(P_h, P_g) \frac{C(P_h)}{C(P_g)}, \quad (6)$$

где  $k^c(P_h, P_g)$  – коэффициент перерасчета производительности различных типов процессоров одинаковой частоты,  $C(P_h)$  и  $C(P_g)$  – значения тактовых частот этих процессоров. Коэффициент  $k^c(P_h, P_g)$  зависит от архитектуры обрабатывающих центров и используемых в них процессорах. Эффективность работы с памятью учитывается с помощью коэффициента  $k^m(P_h, P_g)$ .

Для корректной работы алгоритма управления ресурсоемкими задачами необходимо разработать функциональную модель оценивания загруженности РВС [7-9].

Функциональная модель оценивания загруженности элементов РВС определяется с помощью следующего выражения:

$$M_0 = \{M_M, M_{pz}, M_{np}\}, \quad (7)$$

где  $M_M$  – подсистема мониторинга, осуществляющая сбор контролируемых параметров РВС  $\{q_i\} \in Q$ , где  $Q$  – множество контролируемых параметров;  $M_{pz}$  – подсистема расчета степени загрузки элементов гетерогенной системы;  $M_{np}$  – подсистема принятия решения о загруженности элементов РВС.

Можно выделить два метода получения информации подсистемой  $M_M$  о текущем состоянии  $i$ -го обрабатывающего центра – внешнего и внутреннего мониторинга. При внешнем мониторинге система оценивания загруженности собирает данные о состоянии всех обрабатывающих центров РВС. Система оценивания рассчитывает время отклика Оц[10]. Если система оценивания не получает отклика от обрабатывающего центра после нескольких

последовательных запросов, то данный узел считается недоступным и исключается из списка доступных. Данный метод позволяет получить данные о загрузке ОЦ, выраженные только во времени ответа на запрос системы оценивания загруженности. Такие существенные параметры обрабатывающего центра РВС, как состояние процессора, памяти или системы ввода/вывода, с помощью внешнего мониторинга получить нельзя. Кроме того, большое время ответа обрабатывающего центра может и не зависеть от его загруженности, а являться следствием каких-либо перегрузок среды передачи данных .

Для получения более детальной и объективной информации о состоянии обрабатывающих центров и отдельных их компонентов необходимо использовать метод внутреннего мониторинга, при котором на каждый узел РВС помещается программа-агент, собирающая данные об узле, на котором она находится. Контролируемые параметры через определенный интервал времени передаются системе оценивания загруженности.

Подсистема  $M_{pz}$  производит расчет степени загрузки  $i$ -го обрабатывающего центра в некотором цикле управления. На основании полученных результатов подсистема  $M_{np}$  анализирует их и передает оценку загруженности  $i$ -го элемента РВС вышестоящей системе управления.

В соответствии с методологией функционального моделирования IDEF0 процесс оценивания загруженности РВС можно представить в виде функционального блока, показанного на рисунке, входами которого являются контролируемые параметры (параметры загруженности элементов РВС), полученные с помощью подсистемы мониторинга, а выходами – значения оценок загруженности, полученные подсистемой принятия решения о загруженности элементов РВС.

Управляющим воздействием в данной схеме является методика получения результата оценивания загруженности элементов РВС, а

---

ресурсами – доступные вычислительные ресурсы, используемые для получения результата.

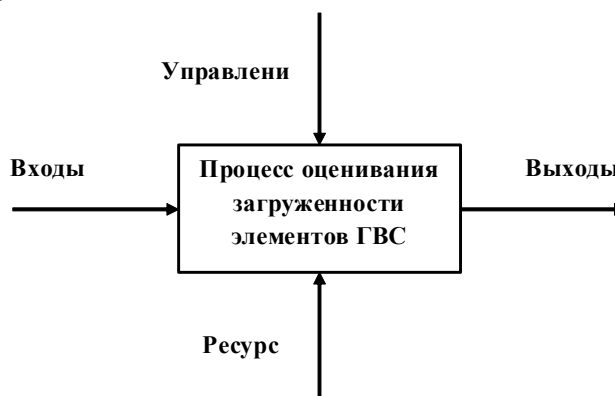


Рисунок- Функциональный блок процесса оценивания в нотации IDEF0

Рассмотрим функционирование подсистемы  $M_{pz}$  более подробно. Для оценки степени загрузки обрабатывающего центра РВС введем понятие единицы загрузки ОЦ, определенное как сумма оценок ресурсов различного вида, выделяемых для выполнения задачи. К таким ресурсам можно отнести время работы процессора данного типа  $T_i$ , число операций ввода/вывода ( $H_i$ ), объем оперативной памяти ( $S_i$ ).

Под единицей процессорного времени будем понимать среднее время выполнения данным типом процессора некоторого фиксированного числа операций. Под единицей ввода/вывода понимается время выполнения одной операции ввода/вывода при вводе/выводе некоторого фиксированного объема информации, не превышающего заранее определенную величину. Единица оперативной памяти представляет некоторый фиксированный объем оперативной памяти, используемый в единицу времени.

В этом случае число единиц обслуживания для некоторой задачи  $i$  за время  $T_i$  при заданном числе операций ввода/вывода  $H_i$  и требуемом объеме памяти  $H_i$  определяется выражением:

$$X_i(T_i, I_i, S_i) = a \frac{T_i}{T'} + \beta I_i + \gamma \frac{S_i}{S'}, \quad (8)$$

где  $\alpha, \beta, \gamma$  – коэффициенты, выбираемые в зависимости от характеристик обрабатывающего центра РВС. Таким образом, изменяя

данные коэффициенты, можно менять «веса» каждой составляющей единицы обслуживания.

Определенное выше понятие единицы загрузки дает возможность оценки степени загруженности каждого ОЦ РВС, а также всей РВС в целом. Максимальная загрузка обрабатывающего центра будет оцениваться за некоторый интервал времени  $T$ , при некотором числе операций ввода/вывода  $I$  за этот интервал, используя весь объем оперативной памяти  $S$  следующим образом:

$$X(T, IS) = a \frac{T}{T'} + \beta I + \gamma \frac{S}{S'} \quad (9)$$

Исходя из определенных выше понятий необходимое число единиц обслуживания для выполнения заданного числа задач  $N$  за некоторое время  $T$  описывается выражением:

$$\sum_{i=1}^N X_i(T_i I_i S_i), \sum_i T_i < T \quad (10)$$

Соответственно загрузка всей системы, состоящей из  $J$  обрабатывающих центров, может быть представлена с помощью выражения :

$$\sum_i^j \sum_i^N X_i(T_i I_i S_i), \sum_i^j \sum_i^N T_i < T \cdot J \quad (11)$$

Воспользуемся стохастической оценкой загрузки обрабатывающего центра вычислительной системы и оценим вероятность того, что процессор будет простаивать, т.е. степень его загрузки равна 0.

Допустим, что длительность обмена информацией между каждой парой ОЦ системы является некой случайной величиной, имеющей экспоненциальное распределение с параметром  $\lambda$ . Таким образом, если узел передает или принимает информацию, то вероятность того, что он закончит передавать информацию до момента  $t+h$ , описывается выражением:

$$P_1(h) = \lambda h e^{-\lambda h}. \quad (12)$$

В частности, при малых  $h$  имеет место выражение :

$$P_1(h) = \lambda h + O(h). \quad (13)$$

Пусть обработка процессором единицы загрузки занимает случайное время, распределенное по экспоненциальному закону с параметром  $\mu$ . Таким образом, в случае работы процессора в момент времени  $t$ , вероятность переработки данной единицы загрузки до момента  $t+h$  при малых значениях  $h$  описывается выражением (14):

$$P(h) = \mu h + O(h). \quad (14)$$

Процессор обрабатывающего центра вычислительной системы может находиться в одном из следующих состояний:  $S_0$  – процессор простаивает, а все каналы работают на обмен;  $S_1$  – один канал системы ожидает окончания обработки его информации, процессор узла системы работает; ... ;  $S_n$  –  $n$  каналов стоят в очереди на обработку, процессор работает; ... ;  $S_N$  – все  $N$  каналов ожидают в очереди на обработку, процессор работает.

Переход из состояния  $S_n$  в состояние  $S_{n+1}$  ( $n+1$  каналы стоят в очереди на обработку) происходит по окончании передачи информации одним из каналов системы. Переход из состояния  $S_n$  в состояние  $S_{n-1}$  означает, что процессор окончил переработку очередной части информации. Таким образом, получается случайный процесс переходов из одного состояния в другое. Этот процесс можно характеризовать вероятностями  $P_n(t)$  того, что в момент времени  $t$  система находится в состоянии  $S_n$ .

Вероятность того, что в момент  $t+h$  будет состояние  $S_0$ , т.е. процессор будет простаивать, равна сумме вероятностей двух событий:

1. В момент  $t$  процессор простаивал и за время  $h$  ни один из каналов системы не закончил с ним обмена:

$$P_{00}(t+h) = P_0(1 - N\lambda h) + O(h). \quad (15)$$

2. В момент  $t$  процессор работал и за время  $h$  окончил работу, а каналы системы не закончили с ним обмен :

$$P_{10}(t+h) = P_1(t)(1 - N\lambda h)\mu h + O(h) = P_1(t)\mu h + O(h). \quad (16)$$

Следовательно, получим выражение:

$$P_0(t+h) = P_0(t) - P_0(t)N\lambda h + P_1(t)\mu h + O(h). \quad (17)$$





$$P_{N-m} = \frac{\frac{1}{m!} \left(\frac{\mu}{\lambda}\right)^m}{1 + \sum_{k=0}^N \frac{1}{k!} \left(\frac{\mu}{\lambda}\right)^k}. \quad (24)$$

Таким образом, вероятность того, что процессор будет простаивать, определяется выражением (25):

$$P_{N-m} = \frac{\frac{1}{N!} \left(\frac{\mu}{\lambda}\right)^N}{\sum_{k=0}^N \frac{1}{k!} \left(\frac{\mu}{\lambda}\right)^k}. \quad (25)$$

Значение  $P_0$  позволяет оценить возможность процессора выполнить дополнительно вновь поступившую задачу.

### Заключение

Формализована задача повышения эффективности диспетчеризации задач в распределенных вычислительных системах. Для этого были определены совокупность моделей, описывающих структуру, ресурсы и входной поток задач (процессов) распределенных вычислительных систем, а также критерии эффективности плана распределения ресурсов. Разработана функциональная модель оценивания загруженности распределенных вычислительных систем, состоящая из подсистемы мониторинга, осуществляющей сбор контролируемых параметров РВС, подсистемы расчета степени загрузки элементов РВС, подсистемы принятия решения о загруженности элементов РВС. Анализ поступающих на выполнение задач в РВС показал, что критерий оценивания загруженности РВС, основанный на определении вероятности простоя вычислительного узла, является достаточно объективным.

*Работа выполнялась при финансовой поддержке РФФИ (договор № 19-07-00079\19).*

### Литература

1. Горбачев Д.В. Инфраструктурные решения распределенных

вычислительных систем. Обзор технологий. Интеллект. Инновации. Инвестиции. 2010. № 4. С. 199-205.

2. Гагарина Л.Г., Румянцева Е.Л., Ен Тве. Характеристики распределенных систем и их классификация. Оборонный комплекс - научно-техническому прогрессу России. 2007. № 4. С. 34-36.

3. Квач А.И., Портнов Е.М. Методика балансировки нагрузки в системах управления IP-шлюзами. Естественные и технические науки. 2018. № 7 (121). С. 157-159.

4. Высочкин А.В., Кокин В.В., Портнов Е.М. Математическая модель оценки вычислительных ресурсов. Международная научно-техническая конференция “Энергосбережение и эффективность в технических системах”. Тамбовский государственный технический университет, 2017. С.321-322.

5. Шмидт. Д. Управляющие системы и автоматика. М. Техносфера, 2007. С. 584.

6. Abzeldin Adamov. Distributed file system as a basis of data-intensive computing. 2012 6th International Conference on Application of Information and Communication Technologies (AICT). С.1-3.

7. Рапопорт Э.Я. Оптимальное управление системами с распределенными параметрами.- М.: Высшая школа, 2009. С. 677.

8. Филатов В.А., Козырь О.Ф. Модель поведения автономного сценария в задачах управления распределенными информационными ресурсами // Инженерный вестник Дона, 2013, №3. URL: [ivdon.ru/ru/magazine/archive/n3y2013/1771](http://ivdon.ru/ru/magazine/archive/n3y2013/1771).

9. Darek Bober, Henryk Kapron, Distributed System for Data Acquisition and Management of Electric Energy Consumption. IEEE IDAACS 2009; Rende, Cosenza, Italy, 21 - 23 Sept. 2009. pp. 192-195.

10. Гибадуллин И.А. Методика оценки производственных комплексов электроэнергетики // Инженерный вестник Дона. 2014, №2. URL:

---



ivdon.ru/ru/magazine/archive/n2y2014/2335.

### References

1. Gorbachev D.V. Infrastrukturnye resheniya raspredelennykh vychislitel'nykh sistem. Obzor tekhnologiy. 2010. № 4. pp. 199-205.
2. Gagarina L.G., Rummyantseva E.L., En Tve Kharakteristiki raspredelennykh sistem i ikh klassifikatsiya. 2007. № 4. pp. 34-36.
3. Kvach A.I., Portnov E.M. Metodika balansirovki nagruzki v sistemakh upravleniya IP-shlyuzami. 2018. № 7 (121). pp. 157-159.
4. Vysochkin A.V., Kokin V.V., Portnov E.M. Matematicheskaya model' otsenki vychislitel'nykh resursov. 2017. pp. 321-322.
5. Shmidt, D. Upravljajushhie sistemy i avtomatika. [Control systems and automation]. 2007. pp. 584.
6. Abzetdin Adamov. Distributed file system as a basis of data-intensive computing. pp. 1-3.
7. Rapoport Je.Ja. Optimal'noe upravlenie sistemami s raspredelennymi parametrami [Optimal management of distributed parameter systems] 2009. pp. 677.
8. Filatov V.A., Kozyr' O.F. Inzhenernyj vestnik Dona, 2013, №3. URL: [ivdon.ru/ru/magazine/archive/n3y2013/1771](http://ivdon.ru/ru/magazine/archive/n3y2013/1771).
9. Darek Bober, Henryk Kapron, Distributed System for Data Acquisition and Management of Electric Energy Consumption. 2009. pp. 192-195.
10. Gibadullin I.A. Inzhenernyj vestnik Dona, 2014, №2. URL: [ivdon.ru/ru/magazine/archive/n2y2014/2335](http://ivdon.ru/ru/magazine/archive/n2y2014/2335).