УДК 004.9

ДИСКРЕТНАЯ МАРКОВСКАЯ МОДЕЛЬ КЛАСТЕРА С СОВМЕСТНЫМ ИСПОЛЬЗОВАНИЕМ ДИСКОВОГО ПРОСТРАНСТВА

Фельдман Л.П., Юсков А.Г.
Донецкий национальный технический университет
кафедра прикладной математики и информатики
E-mail: feldman@r5.dgtu.donetsk.ua, usatank@vnet.dn.ua

Аннотация

Фельдман Л.П., Юсков А.Г. Дискретная марковская модель кластера с совместным использованием дискового пространства. *Предлагается марковская модель кластера с постоянным количеством задач, которая обобщает ранее моделированные модели аналогичных кластеров.*

Введение

В современном мире существует большое количество задач, требующих применения мощных вычислительных систем (ВС).

Одним из примеров вычислительных систем с распределенной обработкой являются кластеры [1,2,3]. Кластерные вычислительные системы по критерию совместного использования дискового пространства классифицируются следующим образом: с совместным использованием дискового пространства и без предоставления доступа к ресурсам [4].

При проектировании и эксплуатации распределенных ВС возникает проблема рационального использования ресурсов вычислительной среды. Для эффективного решения этой проблемы используются непрерывные [5,6,7] или дискретные аналитические модели [8], и для каждого класса решаемых задач определяются основные параметры вычислительной среды.

Непрерывные модели являются менее трудоемкими и применяются для сложного класса исследуемых структур. Дискретная модель Маркова, в сравнении с непрерывной – более точно отражает работу вычислительной среды и позволяет эффективно распараллелить вычислительные структуры [9].

Анализ кластерных систем с помощью дискретной модели при большом количестве решаемых задач на ВС требует больших временных затрат, так как количество состояний дискретной Марковской модели комбинаторно возрастает при увеличении количества задач. В работе представлена марковская модель кластера с совместным использованием дискового пространства, которая является обобщением модели, представленной в [9].

Дискретная модель кластера с совместным использованием дискового пространства

В данной модели каждый узел в кластере имеет свою собственную память, все узлы используют дисковую подсистему. Узлы в кластере обращаются к дисковой подсистеме по системной шине. Вычислительные задачи равномерно распределяются по нескольким объединенным в сеть компьютерам.

В упрощенной структуре кластера с совместным использованием дискового пространства (см. рис. 1) время передачи данных по шине включили во время обслуживания на серверах и дисках, соответственно. Структура состоит из M рабочих станций пользователей, N1 серверов, выполняющие различные приложения, N2 дисков, на которых хранится база данных.

Представим серверы и диски приборами, время обслуживания которых имеет геометрическое распределение со средним параметром q_i (i=1,…,N1+N2+1) – вероятностью завершения обслуживания заявки (соответственно, r_i=1- q_i – вероятность продолжения обслуживания заявки).

Управляющий сервер распределяет пользовательские заявки каждого из M пользователей и с вероятностью $p_{N1+N2+1,i}$ ($i = \overline{1, N1}$) посылает запрос к одному из N1 серверов,

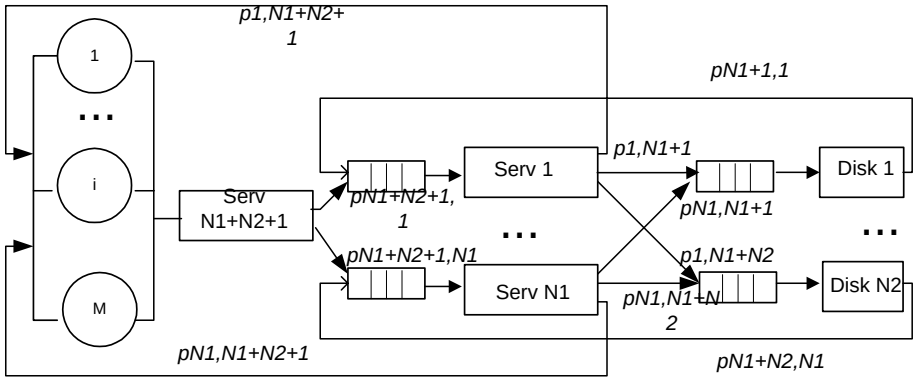

Рисунок 1 – Структурная схема кластера без предоставления доступа к ресурсам

которые, в свою очередь, обрабатывая этот запрос обращаются к одному из N2 дисков с вероятностью p_{ij} ($i = \overline{1, N1}$, $j = \overline{N1+1, N1+N2}$).

Предположим, что рассматриваемая вычислительная система – замкнутая, т.е. в системе обслуживается постоянное количество требований M.

Для построения дискретной модели такой вычислительной системы определим все возможные состояния. За состояние системы примем размещение M заявок по N1+N2+1 узлам $\overline{m} = (m_1,...,m_{N1},m_{N1+1},...,m_{N1+N2},m_{N1+N2+1})$, где m_i – количество задач в i-ом узле (всего узлов N=N1+N2+1). Обозначим множество состояний через $S = \{(m_1,...,m_{N1},m_{N1+1},...,m_{N1+N2},m_{N1+N2+1} | \sum_{i=1}^{N1+N2+1} m_i = M\}$. Число L всевозможных состояний системы равно числу размещений M задач по N узлам и определяется по формуле:

$$L = C_{M+N-1}^{N-1}. \quad (1)$$

Количество устройств в каждом узле задается вектором $\overline{k} = (k_1,...,k_{N1},k_{N1+1},...,k_{N1+N2},k_{N1+N2+1})$.

Определим переходные вероятности для каждой пары состояний $P_{ij}^{(k)} = P\{S_j^{(k)} | S_j^{(k-1)}\}$, т.е. вероятности перехода из состояния S_i, в котором она находилась на (k-1)-м шаге, в состояние S_j на k-м шаге.

Для произвольного состояния $\overline{m}$ введем вектор α, s-я компонента которого

$$\alpha_S = \min(m_S, k_S), \quad s = \overline{1, \mathrm{N1+N2+1}}, \quad (2)$$

определяет число загруженных устройств в s-м узле обработки задач. Для определения возможности перехода из произвольного состояния $\overline{m}$ в произвольное состояние $\overline{m'}$ найдем вектор

$$\bar{i} = \overline{m} - \overline{m'} = (i_1, i_2,..., i_{\mathrm{N1+N2+1}}), \quad (3)$$

каждая компонента i_S которого представляет изменение числа задач в s-ом узле. Если i_S>0, то на рассматриваемом переходе из узла s должны уйти минимум i_S задач. Если i_S <

0, то в узел s должны прийти $|i_S|$ задач. При любом переходе из $\overline{m}$ число задач, обслуженных s-м узлом обработки, не может быть больше α_s, т.е.

$$i_S \le \alpha_S, \quad s = \overline{1, \mathrm{N1}+\mathrm{N2}+1}. \tag{4}$$

Определим множество J номеров узлов обработки, в которых $i_S < 0$, т.е.

$$J = \{S\big|_{i_S<0}\}, \quad s = \overline{1, \mathrm{N1}+\mathrm{N2}+1}. \tag{5}$$

Если $J \neq \{\varnothing\}$, то величины $\gamma_S = |i_S|$ $(S \in J)$ определяют минимально возможное число программ, которые поступают к тем узлам обработки, номера s которых принадлежат множеству *J*. Число программ, поступивших в один из таких узлов s равно $|i_S|$. Количество поступивших задач к главному серверу от серверов не может превышать количества работающих серверов

$$|i_{N1+N2+1}| \le \sum_{j=1}^{N1} \alpha_j . \tag{6}$$

Аналогично, количество поступивших задач к дискам от серверов не может превышать количества работающих серверов

$$\sum_{j=N1+1}^{N1+N2} |i_j| \le \sum_{j=1}^{N1} \alpha_j , j \in J . \tag{7}$$

От серверного узла задачи поступают и к управляющему серверу, и к дискам, поэтому общее количество ушедших задач не превышает количества работающих серверов

$$|i_5| + \sum_{j=N1+1}^{N1+N2} |i_j| \le \sum_{j=1}^{N1} \alpha_j , j \in J . \tag{8}$$

Количество поступивших задач к серверам не превышает количества работающих дисков и задачи, поступившей от управляющего сервера

$$\sum_{j=1}^{N1} |i_j| \le \sum_{j=N1+1}^{N1+N2} \alpha_j + 1, j \in J . \tag{9}$$

Если $J = \{\varnothing\}$, то $\gamma_S = 0$.

Рассмотрим частный случай:

N = 5 – количество устройств,

N1 = 2 – количество вычислительных серверов

N2 = 2 – количество дисков

M = 5 – количество задач.

Определим число L всевозможных состояний системы:

$$L = C_{M+N-1}^{N-1} = C_6^4 = 126 .$$

Переход $\overline{m} - \overline{m'}$ возможен, если соблюдаются условия (6-9).

Рассмотрим случай $i_5=1$ – управляющий сервер завершил обработку одной задачи, следовательно, обработка требуется на одном из вычислительных серверов.

Допустим, другие устройства продолжают обработку задач, и обмен не требуется, тогда вероятность перехода между двумя состояниями:

$$P_{\overline{m} \to \overline{m'}} = q_5 P_{5s} \prod_{l=1}^{4} C_{\alpha_l}^{i_l} q_l^{i_l} r_l^{\alpha_l - i_l} \tag{10}$$

Рассмотрим второй вариант, когда $i_5=1$ – управляющий сервер завершил обработку одной задачи, другие устройства также завершили обработку задач, следовательно, необходимо учесть возможность обмена задачами между вычислительными серверами и дисковыми устройствами.

Сформулируем условия формирования множества для обмена задачами:

$$I_{s1} = \{s\big|_{\alpha_s>0}\}, s = \overline{1,2} \quad (11)$$

$$I_{s2} = \{s\big|_{\alpha_s>0}\}, s = \overline{3,4} \quad (12)$$

Множество I_{s1} содержит номера вычислительных серверов, а множество I_{s1} – номера дисков, которые в данном такте могут завершить обработку задачи.

Обмен может быть осуществлен при условии (13):

$$\left|I_{s1}\right| \neq \{\varnothing\} \wedge \left|I_{s2}\right| \neq \{\varnothing\} \quad (13)$$

Построим множество событий для обмена задачами:

$$I = I_{s1} \times I_{s2} \quad (14)$$

$$\left|I\right| = \left|I_{s1}\right| \cdot \left|I_{s2}\right| \quad (15)$$

Элемент множества I имеет следующий вид:

$$\{l_1, l_2\}, \quad (16)$$

где $l1$ - номер серверного узла,

$l2$ - номер дискового узла.

Тогда вероятность перехода между двумя состояниями:

$$P_{\overline{m}\to\overline{m'}} = q_5 P_{5s} \prod_{l=1}^{4} C_{\alpha_l}^{i_l} q_l^{i_l} r_l^{\alpha_l - i_l} \cdot \sum_{l=1}^{|I|} C_{\alpha_{l_1}}^{i_{l_1}} q_{l_1}^{i_{l_1}} r_{l_1}^{\alpha_{l_1} - i_{l_1}} P_{l1,l2} C_{\alpha_{l_2}}^{i_{l_2}} q_{l_2}^{i_{l_2}} r_{l_2}^{\alpha_{l_2} - i_{l_2}} P_{l2,l1} \quad (17)$$

Рассмотрим случай i_5=0, возможны три варианта обмена задачами:

1) между управляющим сервером и вычислительными серверами;
2) между вычислительными серверами и дисковыми устройствами;
3) между управляющим сервером и вычислительными серверами, а также вычислительными серверами и дисковыми устройствами.

Сформулируем условия формирования множества для обмена задачами:

$$I_{s1} = \{s\big|_{\alpha_s>0}\}, s = 5 \quad (18)$$

$$I_{s2} = \{s\big|_{\alpha_s>0}\}, s = \overline{1,2} \quad (19)$$

$$I_{s3} = \{s\big|_{\alpha_s>0}\}, s = \overline{3,4} \quad (20)$$

Множество I_{s1} содержит номер управляющего сервера, I_{s2} – номера вычислительных серверов, а множество I_{s3} – номера дисков, которые в данном такте могут завершить обработку задачи.

Обмен может быть осуществлен при условии (21):

$$\left|I_{s1}\right| \neq \{\varnothing\} \wedge \left|I_{s2}\right| \neq \{\varnothing\} \vee \left|I_{s2}\right| \neq \{\varnothing\} \wedge \left|I_{s3}\right| \neq \{\varnothing\} \vee \left|I_{s1}\right| \neq \{\varnothing\} \wedge \left|I_{s2}\right| \neq \{\varnothing\} \wedge \left|I_{s3}\right| \neq \{\varnothing\} \quad (21)$$

Построим множество событий для обмена задачами:

$$I = I_{s1} \times I_{s2} \times I_{s3} \quad (22)$$

$$\left|I\right| = \left|I_{s2}\right| \cdot \left|I_{s3}\right| \quad (23)$$

Элемент множества I имеет следующий вид:

$$\{l_1, l_2, l_3\}, \quad (24)$$

где $l1$ – номер управляющего сервера,

$l2$ – номер серверного узла,

$l3$ – номер дискового узла.

Для определения вектора стационарных состояний надо решить систему линейных алгебраических уравнений (СЛАУ)

$$\overline{\pi}(k) = \overline{\pi}P, \quad (25)$$

соответствующую рассмотренной марковской модели.

Основные характеристики моделируемой ВС определяются с использованием стационарных вероятностей [5].

Среднее число занятых устройств в s-м узле определяется по формуле:

$$k_s^{cp} = k_s - \sum_{l \in As} (k_s - m_s(l))\pi_l. \quad (26)$$

Загрузка устройств определяется по следующей формуле:

$$\rho_s = k_s^{cз} / k_s. \quad (27)$$

Среднего числа задач, находящихся в s-м узле:

$$m_s^{cp} = \sum_{l=1}^{L} m_s(l)\pi_l \quad (28)$$

Среднее число задач, находящихся в очереди к s-му узлу:

$$l_s^{cp} = m_s^{cp} - \rho_s k_s \quad (29)$$

Выводы

Предложена обобщенная модель кластерной вычислительной системы с совместным использованием дискового пространства, которая может использоваться для повышения качества работы вычислительного кластера.

Алгоритм построения элемента матрицы переходных вероятностей и определения вектора стационарных вероятностей позволяет их легко распараллелить на многопроцессорные вычислительные структуры.

Список литературы

1. Шнитман В. Современные высокопроизводительные компьютеры. Информационно-аналитические материалы центра информационных технологий, 1996: http//hardware/app_kis.

2. Corbalan J., Martorell X., Labarta J. Performance-Driven Processor Allocation //IEEE Transactions on Parallel and Distributed Systems, vol. 16, No. 7, July 2005, PP.599-611

3. Oleszkiewicz J., Xiao L., Liu Y. Effectively Utilizing Global Cluster Memory for Large Data-Intensive Parallel Programs //IEEE Transactions on Parallel and Distributed Systems, vol. 17, No. 1, Jan. 2006, PP.66-77

4. Спортак М., Франк Ч., Паппас Ч. и др. Высокопроизводительные сети. Энциклопедия пользователя.-К.: «ДиаСофт», 1998.-432с.

5. Авен О. И. и др. Оценка качества и оптимизация вычислительных систем. – М.: Наука, 1982, 464с.

6. Cremonesi P., Gennaro C. Integrated Performance Models for SPMD Applications and MIMD Architectures //IEEE Transactions on Parallel and Distributed Systems, vol. 13, No. 7, Jul. 2002, PP.745-757

7. 7 Varki E. Response Time Analysis of Parallel Computer and Storage Systems //IEEE Transactions on Parallel and Distributed Systems, vol. 12, No. 11, Nov. 2001, PP.1146-1161

8. Клейнрок Л. Вычислительные системы с очередями. – М.:Мир, 1979, 600с. Последовательно - параллельные вычисления: Пер. с англ. - М.: Мир, 1985. - 456 с.

9. Фельдман Л.П., Михайлова Т.В. Параллельный алгоритм построения дискретной марковской модели /Высокопроизводительные параллельные вычисления на кластерных системах. Материалы четвертого Международного научно-практического семинара и Всероссийской молодежной школы. /Под редакцией член-корреспондента РАН В.А. Сойфера. – Самара, 2004. – С. 249–255.