УДК 004.021

ИССЛЕДОВАНИЕ ПАРАЛЛЕЛЬНОГО АЛГОРИТМА ПОСТРОЕНИЯ МАРКОВСКОЙ МОДЕЛИ ВЫЧИСЛИТЕЛЬНОГО КЛАСТЕРА

Дяченко Т.Ф., Михайлова Т.В.
Донецкий национальный технический университет

В работе предложен параллельный алгоритм построения Марковской модели вычислительного кластера с совместным использованием дискового пространства. Приведены теоретические оценки трудоемкости последовательного алгоритма и параллельного, отображенного на решетку процессоров.

5

Начиная с момента появления вычислительной техники, человек постоянно стремится улучшить ее производительность, т.е. повысить количество обрабатываемой информации в единицу времени или сократить временные затраты компьютера на выполнение конкретной программы. Осуществление этой цели достигается двумя путями. Первый – непрерывное совершенствование аппаратных элементов вычислительной техники, сокращение связей между ними, повышение скорости их работы. Другой путь повышения производительности – создание суперкомпьютеров.

Актуальность темы

Анализ способов построения высокопроизводительных кластерных систем и средств оценки их эффективности показывает сложность решения задачи анализа и синтеза этих вычислительных систем. Традиционных, классических решений этой задачи уже недостаточно и возникает необходимость в разработке новых подходов для её решения.

1. Наличие конфликтных ситуаций при использовании общих ресурсов, а также несоответствие программ

архитектуре вычислительных систем (ВС) приводят к снижению производительности вычислительных систем. Для оценки и рекомендации на стадии проектирования следует использовать аналитические методы анализа, а именно, аппарат Марковских цепей.

2. Аналитические методы, в основном, используются для анализа производительности однопроцессорных ВС, преимущественно с последовательной обработкой данных. Для ВС с параллельной обработкой данных и для многопроцессорных ВС, например, многопроцессорных структур, в том числе, кластерных, аналитические методы требуют дальнейшего развития.
3. Дискретные Марковские модели, описывающие проектируемые вычислительные сети, имеют большую размерность и в большинстве случаев не решаются на ЭВМ ввиду ограниченности вычислительных ресурсов. Одним из способов решения этой проблемы является распараллеливание этих алгоритмов.
4. Большое значение для проектирования ВС имеют не только задача анализа, но и задача синтеза. Однако для современных вычислительных структур, когда в системах «клиент-сервер» количество клиентов достигает сотни, задачи синтеза не разрешаются ввиду большой размерности и нелинейности системы уравнений, получаемой при решении задачи.

Нахождение способа решения таких задач позволит сократить процесс проектирования ВС большой размерности.

Построение Марковской модели однородного вычислительного кластера

Одним из примеров вычислительных систем с распределенной обработкой являются кластеры. Они обладают высокой производительностью и «живучестью» вычислительных ресурсов. Классификация кластерных систем по совместному использованию одних и тех же дисков отдельными узлами является самой распространенной [1, 2]. Одной из основных проблем при построении ВС с распределенной обработкой является выработка рекомендации для рационального использования ресурсов этой сети. Эта задача возникает как при эксплуатации, так и при проектировании сети. Для эффективной эксплуатации кластерной системы необходимо построить её аналитическую модель и для каждого класса решаемых задач определить основные параметры вычислительной среды. Эти модели можно использовать и для оптимизации ВС на стадии проектирования.

В данной работе рассматривается кластер с совместным использованием дискового пространства. Такие топологии используют СУБД Oracle Parallel Server и Informix [2]. Производительность таких систем может увеличиваться как путем наращивания числа процессоров и объемов оперативной памяти в каждом узле кластера, так и посредством увеличения количества самих узлов.

Принципиальная схема кластера такого типа приведена на рис. 1. Упрощенная модель кластера с совместным использованием дискового пространства приведена на рис. 2

Входной узел (управляющий сервер) равномерно распределяет между серверами приложений (выполняющими одинаковые приложения) задачи. Количество серверов – приложений — N1. Каждый из них может обратиться к данным, расположенным

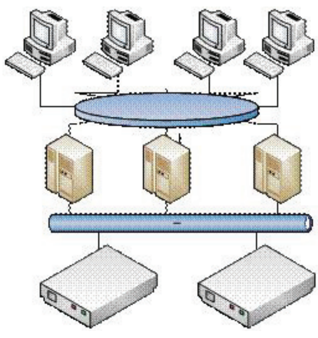

Рисунок 1 – Архитектура кластера с совместным использованием дискового пространства

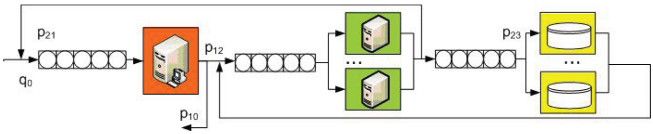

Рисунок 2 – Структурная схема Марковской модели кластера с совместным использованием дискового пространства

на дисках, количество которых N2. Ввиду ограниченных вычислительных возможностей будем считать, что количество задач, обрабатываемое такой вычислительной системой не более M.

Построим дискретную Марковскую модель. Так как кластер однородный, представим серверы, и, аналогично, дисковые накопители многоканальными устройствами. Требования, поступающие на обслуживание на серверы, поступают в ограниченную очередь (не более M), из которой на обслуживание выбираются по правилу «первый пришел – первый обслужился».

Допустим, задачи, обрабатываемые на таком кластере, однородные и имеют следующие характеристики:

p12 – вероятность запроса к одному из N1 серверов,

p23 – вероятность запроса к одному из N2 дисков,

p21 – вероятность завершения обслуживания одним из N1 серверов,

p10 – вероятность завершения обслуживания задачи,

q0 – вероятность появления новой задачи.

При построении модели используется методика, предложенная в [3].

Алгоритм построения дискретной Марковской модели состоит из двух частей: вычислений матрицы переходных вероятностей и вектора стационарных вероятностей.

При определении матрицы переходных вероятностей используются следующие параметры: N – количество узлов в моделируемой системе (устройств или их блоков), M – количество заявок в системе.

Число состояний системы равно числу размещений j задач по

N узлам и определяется по формуле:

$$L_j = C_{j+N-1}^{N-1},$$

общее количество состояний вычисляется так:

$$L = \sum_{j=0}^{M} L_j.$$

Общее количество ненулевых элементов матрицы переходных вероятностей вычисляется как:

$$K_1 = N + \sum_{j=1}^{M-1} C_{j+N-1}^{N-1}(C_{j+N-2}^{N-1} + C_{j+N-1}^{N-1} + C_{j+N}^{N-1}) + C_{M+N-1}^{N-1}(C_{M+N-1}^{N-1} + C_{M+N-2}^{N-1}).$$

В таблице 1 приведены результаты расчетов доли ненулевых элементов в матрице переходных вероятностей для модели с тремя группами устройств N=3.

Размерность получаемых матриц зависит от количества задач в моделируемой системе.

Порядок возникающих матриц – от десятков и сотен тысяч до десятков миллионов. Это вызвано желанием оперировать более точными дискретными моделями, позволяющими получать приближенные решения, более близкие к решениям исходных задач, лучше учитывать локальные особенности рассматриваемого процесса.

Таблица 1 – Доля ненулевых элементов в матрице переходных вероятностей

Количество задач M			
20	50	100	200
Размерность матрицы переходных вероятностей (количество состояний модели)			
1771 Ч 1771	23426 Ч 23426	176851 Ч 176851	1373701 Ч 1373701
Количество ненулевых элементов			
1257408	94113953	2746496028	83871314553
Доля ненулевых элементов в матрице переходных вероятностей			
0.4009	0.1715	0.0878	0.0444

Разработка параллельного алгоритма построения Марковской модели

Построение подобной модели требует значительных вычислительных ресурсов, которые могут быть представлены параллельными компьютерами. Поэтому является актуальной проблема создания параллельных алгоритмов решения данной задачи.

Алгоритм построения дискретной Марковской модели состоит из двух частей: вычислений матрицы переходных вероятностей и вектора стационарных вероятностей.

На рис. 3 приведена схема построения Марковской модели с учетом распараллеливания подзадач.

Теоретическая оценка трудоемкости последовательного алгоритма вычисления матрицы переходных вероятностей, учитывая, что tсл ≈ tумн, определяется как

$$T1_{посл} = L \times L \times 2(N+1)\, t_{оп} + K_1 \times 3N\, t_{оп} + K_1 \{(2N + 4k_2 + \sum_{s=1}^{N} k_s - 4) C_{k_2+N-2}^{N-2} \}\, t_{оп},$$

где: L – количество состояний марковской модели;

N – количество узлов вычислительного кластера;

k_s – количество устройств в s-ом узле, $s=1..N$;

K_1 – общее количество ненулевых элементов для всей матрицы переходных вероятностей.

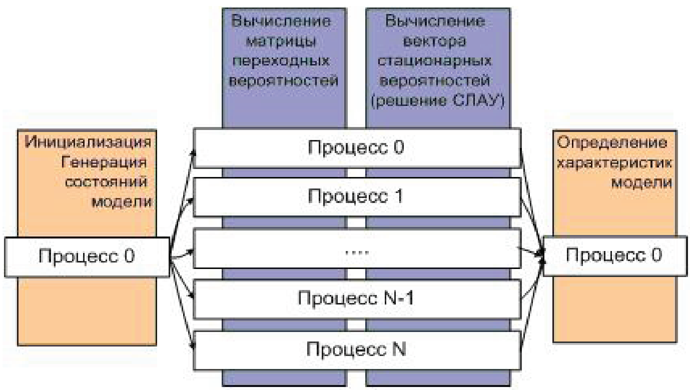

Рисунок 3 – Построение Марковской модели (выделение подзадач)

Отображение параллельного алгоритма на решетку процессоров

Пусть процессор имеет топологию- решетку процессорных элементов (ПЭ) $p=n^2$ типа двумерной прямоугольной решетки-тора. Каждый ПЭ может выполнить любую операцию за один такт; время обращения к запоминающему устройству ПЭ пренебрежительно мало по сравнению со временем выполнения операции на ПЭ.

На каждом ПЭ с индексами *i, j* вычисляется один элемент матрицы переходных вероятностей p_{ij}. Временная сложность на одном ПЭ для вычисления одного элемента матрицы переходных вероятностей определяется как

$$T1_{парал}=(5N+2)\,t_{on}+(2N+4k_2+\sum_{s=1}^{N}k_s-4)C_{k_2+N-2}^{N-2}\quad t_{on}$$

Количество итераций для вычисления матрицы переходных вероятностей вычисляется как

$$\{[\frac{K_1}{p}]+1\}.$$

В этом случае временная сложность параллельного алгоритма вычисления матрицы переходных вероятностей на р процессорах вычисляется следующим образом:

$$T1_{парал}=([\frac{K_1}{p}]+1)\{(5N+2)\,t_{on}+(2N+4k_2+\sum_{s=1}^{N}k_s-4)C_{k_2+N-2}^{N-2}\quad t_{on}\}$$

Ускорение параллельного алгоритма вычисления матрицы переходных вероятностей на решетке процессоров:

$$\frac{K_1\,(5N+2)\,t_{on}+K_1\,\{(2N+4k_2+\sum_{s=1}^{N}k_s-4)C_{k_2+N-2}^{N-2}\ \}\,t_{on}}{(\left[\frac{K_1}{p}\right]+1)\{(5N+2)\,t_{on}+(2N+4k_2+\sum_{s=1}^{N}k_s-4)C_{k_2+N-2}^{N-2}\quad t_{on}\}}$$

Эффективность параллельного алгоритма вычисления матрицы переходных вероятностей на решетке процессоров с увеличением количества решаемых задач М асимптотически приближается к единице (рис.4), а с увеличением количества

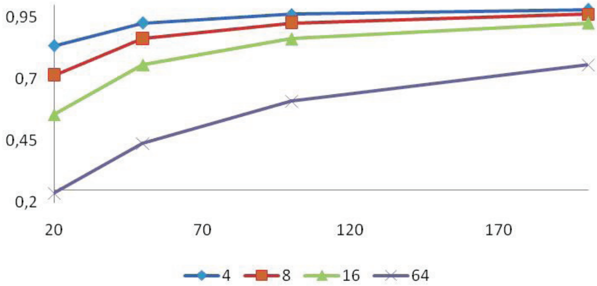

Рисунок 4 – Эффективность параллельного алгоритма вычисления матрицы переходных вероятностей зависимости от количества решаемых задач *M*

процессоров незначительно уменьшается (в сотых долях), что объясняется простаивающими процессорами.

В целом, эффективность параллельного алгоритма вычисления матрицы переходных вероятностей на решетке процессоров более 0.85, в зависимости от количества задач М и количества процессоров p.

5

Литература

[1] Спортак М., Франк Ч., Паппас Ч. и др. Высокопроизводительные сети. Энциклопедия пользователя. – К.: ДиаСофт, 1998. – 432 с.

[2] Цилькер Б.Я., Орлов С.А. Организация ЭВМ и систем – М: ПИТЕР, 2004 – 668 с.

[3] Фельдман Л.П., Малинская Э.Б. Аналитическое моделирование вычислительных систем с приоритетным обслуживанием программ. //Электронное моделирование. – К., 1985. –№6. – С. 78-82.