СПОСОБЫ ОПТИМИЗАЦИИ СОСТАВА И СТРУКТУРЫ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Фельдман Л.П., Михайлова Т.В.

Кафедра ПМиИ ДонГТУ
E-mail: feldman@r5.dgtu.donetsk.ua

Abstract

Feldman L.,Michailova T. Structures optimization methods for high- performance systems. *There is approximation method for synthesis closed queuing systems without calculations error estimation. In this work structure optimization task for systems with a permanent number of tasks is put, the methods of numerical solution with Markov chains are worked.*

Введение

В работе авторов [1] поставлена задача оптимизации состава и структуры вычислительной системы (ВС), представленной замкнутой стохастической сетью, обрабатывающих постоянное число задач, разработан аналитический метод решения этой задачи. В данной работе, являющейся продолжением статьи [1], рассматривается общая постановка задачи оптимизации ВС, представленной замкнутой стохастической сетью, обрабатывающих постоянное число задач, и предлагается более эффективный подход для решения таких задач с использованием градиентных методов.

Марковская модель системы клиент-сервер

Структура системы клиент- сервер приведена на рис.1. В ней M рабочих станций пользователей и N-I серверов. Предполагается, что все рабочие станции заняты пользователями, каждый из которых может послать только один запрос на сервер. Пользователь, пославший запрос к одному из серверов, ждет ответа и, только

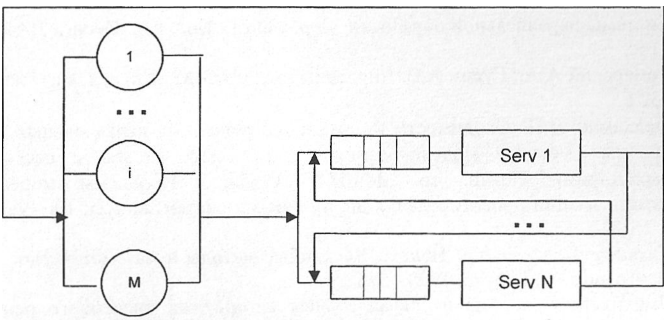

Рисунок 1 Структура системы клиент - сервер

после получения ответа, может послать новый запрос. Таким образом, в системе постоянно находится М запросов, часть из них m i находится у пользователей, а m, - на i-ом сервере (m1+m2+-+mN=M).

Функционирование рассматриваемой системы можно представить замкнутой стохастической сетью, содержащей N систем массового обслуживания (СМО), в которой циркулирует М заявок. Граф передач этой сети изображен на рис.2, на котором введены следующие обозначения:

Si, S2,...,Sk,..., SN - СМО, соответствующие клиентам и серверам;

So- фиктивная система, введенная для фиксации событий завершения задач пользователями;

Pij - вероятности поступлений заявок из i-й СМО в j-ю,

рю- вероятность завершения задачи пользователем.

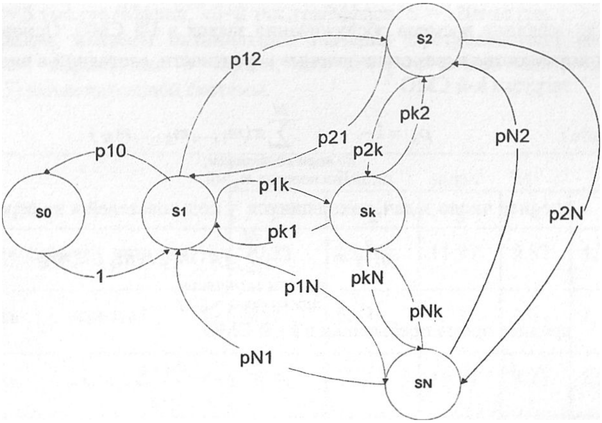

Рисунок 2 Граф передач сети клиент- сервер

Поскольку в ВС находится постоянное число задач, то предполагается, что после завершения очередной задачи пользователь приступает к следующей. На рисунке 2 этому соответствует передача через систему So- По графу передач определяются соотношения интенсивностей потоков заявок, поступающих в каждую из систем

$$\lambda_0 = p_{10}\lambda_1,$$

$$\lambda_2 = p_{12}\lambda_1 + p_{32}\lambda_3 + ... + p_{k2}\lambda_k + ... + p_{N2}\lambda_N,$$

$$..................$$

$$\lambda_k = p_{1k}\lambda_1 + p_{2k}\lambda_2 + ... + p_{k-1,k}\lambda_{k-1} + p_{k+1,k}\lambda_{k+1} + ... + p_{Nk}\lambda_N,$$

$$..................$$

$$\lambda_N = p_{1N}\lambda_1 + p_{2N}\lambda_2 + ... + p_{kN}\lambda_k + ... + p_{N-1,N}\lambda_{N-1}$$

Если разделить все уравнения на $Д_0$ > обозначить через а,· = X_i / Дд. получаются уравнения для определения среднего числа этапов обслуживания задачи пользователем и на сервере соответственно :

$$\alpha_1 = 1/p_{10},$$

$$\alpha_2 = p_{12}\alpha_1 + p_{32}\alpha_3 + ... + p_{k2}\alpha_k + ... + p_{N2}\alpha_N,$$

$$.............$$

$$\alpha_k = p_{1k}\alpha_1 + p_{2k}\alpha_2 + ... + p_{k-1,k}\alpha_{k-1} + p_{k+1,k}\alpha_{k+1} + ... + p_{Nk}\alpha_N,$$

$$............$$

$$\alpha_N = p_{1N}\alpha_1 + p_{2N}\alpha_2 + ... + p_{kN}\alpha_k + ... + p_{N-1,N}\alpha_{N-1}$$

Если за состояние системы принять распределение заявок по СМО (m_b..,m_k,..,m_N), где mi+…+m_k+…+m_N=M, то по теореме Джексона [2] для стационарных вероятностей состояний получим выражение

$$\pi(\mathrm{m}_1,...,\mathrm{m}_k,...,\mathrm{m}_N) = \frac{(\alpha_1 \nu_1)^{m_1}...(\alpha_N \nu_N)^{m_N}}{m_1! \sum\limits_{\text{по всем состояниям}} \dfrac{(\alpha_1 \nu_1)^{m_1}..(\alpha_N \nu_N)^{m_N}}{m_1!}}, \quad (1)$$

где V_i - средние времена обслуживания заявок в i-й СМО. Основные характеристики сети выражаются через стационарные вероятности состояний в виде:

загрузка k-й СМО

$$\rho_k = 1 - \sum_{\substack{\text{по всем состояням,}\\ \text{при котрых } m_k = 0}}^{M} \pi(m_1,...,m_k,...,m_N) \quad (2)$$

среднее число задач, находящихся у пользователей и на серверах

$$m^*_k = \sum_{\substack{\text{по всем состояням,}\\ \text{при котрых } m_k = j}}^{M} j\pi(m_1,...,m_k,...,m_N), \quad (3)$$

среднее время пребывания в к-ой СМО

$$u_k = \frac{\overset{*}{m}_k \nu_k}{\rho_k}, \quad (4)$$

общее время решения задачи

$$U = \sum_{i=1}^{N} \alpha_i u_i \quad (5)$$

Способы оптимизации состава и структуры параллельных высокопроизводительных вычислительных систем и сетей заданной стоимости с минимальным временем ответа

Рассматривается решение следующей задачи: определить быстродействие рабочих станций Vi и серверов V_2 ,---,Vn, обеспечивающих минимальное время решения задачи U так , чтобы стоимость системы с M рабочими станциями не превышала заданного значения S*. Таким образом, необходимо найти минимум функции U, определяемой формулой (5), в которую надо подставить вместо $\boldsymbol{V_k}$ O_k / V_k. При этом надо удовлетворить ограничениям

$$c_1MV_1+c_2V_2+\ldots+c_NV_N \leq S^*, \qquad V_i>0,\ i=1,\ldots N. \quad (6)$$

где C_i- стоимость единицы производительности устройства типа i.

Для решения рассматриваемой задачи можно применить метод множителей Лагранжа, т.е. найти минимум функции

$$G=U+\gamma(c_1MV_1+c_2V_2+\ldots+c_NV_N-S^*),$$

где γ- неопределенный постоянный множитель. В этом случае V^ (k=l,…,N) и γ определяются как решение системы уравнений

$$\frac{\partial G}{\partial V_1}=0,\ldots,\frac{\partial G}{\partial V_k}=0,\ldots,\frac{\partial G}{\partial V_N}=0,\quad c_1MV_1+c_2V_2+\ldots+c_NV_N\le S^*, \tag{8}$$

Выражение для функции U достаточно громоздкое, поэтому для выполнения преобразований, необходимых для получения системы (8), используется пакет *Mathematica.* Программа формирования и решения уравнений (8) подробно описывается в [1].

Результаты численного решения средствами пакета приведены в таблице 1. Расчеты приведены для следующих значений: количество серверов N=2, трудоемкость одного этапа решения задачи в миллионах операций *6[=2,* *θχ=* 0_3 = 2 количество этапов обслуживания а\ =20, «2 =17.8; α_3 =16,6; коэффициенты стоимости cl= 2 тыс.грв/ Мфлоп, с2= 5 тыс.грв /Мфлоп, с3=2 тьгс.грвЛУІфлоп, S = 120тыс.грв.

Таким образом, найдены оптимальные значения быстродействий рабочих станций и серверов, выраженные в Мфлоп, для которых можно найти другие характеристики (2-5) вычислительной системы.

Таблица 1

	Число задач М				
	1	2	3	4	10
Производительность рабочей станции V_1 (Мфл/сек)	20.22	15.25	11.97	9.82	4.79
Производительность сервераI Vj (Мфлоп/сек)	8.54	6.43	5.31	4.61	2.73
Производительность сервера2 V_3 (Мфлоп/сек)	18.44	13.41	10.82	9.23	5.27
Время решения задачи (сек)	5.86	11.7	19.18	28.13	113.62

Время решения этой задачи уже при М=8 способом, приведенным в [1], занимает несколько десятков минут, а при M=IO - больше часа. Это объясняется тем, что три из четырех нелинейных уравнений системы (8), формируемые пакетом *Mathematica,* имеют большой обьем (при М=8 одно уравнение требует 3-5 печатных листа формата А4). Для аналитического решения системы (8) пакет основную часть времени тратит на обмен с диском. Поэтому для решения поставленной задачи (7)-(8) лучше использовать метод штрафных функций и градиентный метод [5].

Для задачи (7)-(8) определяется функция (13), минимум которой находится с помощью градиентного метода.

$$F=U+r^*\,(c_1MV_1+\ldots+c_kV_k+\ldots+c_NV_N-S^*)^2, \tag{13}$$

где $(c_1MV_1+\ldots+c_kV_k+\ldots+c_NV_N-S^*)^2$- «штраф»,

r - весовой коэффициент, численное значение которого требует дополнительных исследований.

Преимущество, которое получается за счет перехода от задачи минимизации функции при условии ограничений к задаче минимизации градиентными методами, состоит в том, что не нужно аналитически решать систему нелинейных уравнений (8), а можно решить численно задачу минимизации функции (13).

Выбирается начальная точка, удовлетворяющая ограничению (6), в которой вычисляется функция (13) и её градиент:

Ll[W_] = U[W_] + γ « (SS - S[W_]) « (SS - S[W_])
vni[i] = vi[i] + ««*aLir
VUl[2] > Vl[2]+oo.dL2;
VUl[3] = Vl[3] + aa.dL3;

Определяется новая точка приближения

dLl=D[LltW_J, VHl]]:
dL2 = D[Ll[W_], Vl[2]];
dL3 = D[Ll[W_], Vl[3]];

Если значение функции (13) в новой точке лучше, чем в предыдущей, осущетвляется переход в новую точку, если - нет, то уменьшается шаг αα в два раза и все повторяется сначала. Поиск минимума прекращается при достижении заданной точности.

Так как функция (5) в заданной области выпукла вниз, что можно увидеть на рисунке 3, минимум её достигается на границе области допустимых решений [6]. Градиентным методом в силу специфичности функции (13) минимум не всегда достигается (при разном исходном приближении или разньгх значениях параметра r получаются разные значения экстремума).

Поэтому методом координатного спуска [5] на границе уточняется результат. Начальным приближением для этого метода используется точка, полученная градиентным методом. Затем на границе с единичным шагом выбирается направление координаты, по которой функция улучшается, и вычисляется новая точка. Если нет лучшей точки, шаг уменьшается. Процесс продолжается, пока не достигнется заданная точность.

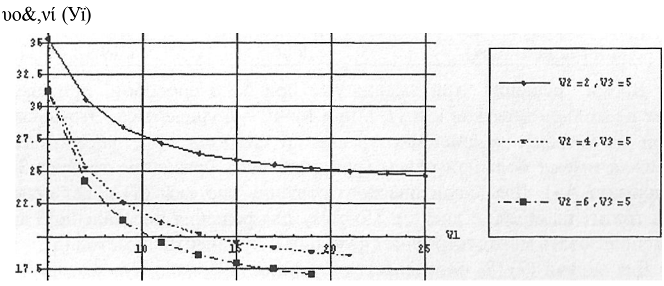

Рисунок 3 Значения функции U при различных переменных

Результаты, полученные методом штрафных функций, совпадают с результатами, приведенными в таблице 1 до первого знака после запятой (норма градиента в полученной точке равна 0.003). Результаты, которые нельзя было получить аналитически, были получены численно и приведены в таблице 2 (M=12, M=14).

Таблица 2

	Число Задач М		
	10	12	14
Производительность рабочей станции V_1 (Мфл/сек)	4.79	4.09	3.56
Производительность сервера! $У_2$(Мфлоп/сек)	2.73	2.46	2.37
Производительность сервера2 V_3 (Мфлоп/сек)	5.27	4.66	4.17
Время решения задачи (сек)	113.62	153.67	200.0

Выводы

Рассмотрены два подхода к задаче синтеза ВС с минимальным временем ответа заданной стоимости: с использованием функции Лагранжа и функций штрафа с уточнением методом покоординатного спуска. Проведенный анализ позволяет сделать вывод о преимуществе второго метода, т.к. время решения задачи существенно отличается в пользу последнегою. Чем больше значение М , тем больше разница по времени решения задачи между обоими методами, а при М>10 ЭВМ не может решить методом функций Лагранжа из-за недостаточного количества системных ресурсов (большое количество обращений к диску). Однако во втором способе имеются и недостатки, например, ввиду вогнутости функции U сложно подбирать начальное приближение, параметр r для функции штрафа, оптимальный шаг для метода покоординатного спуска. Показано [5], что параметр r начинают варьировать с 50 и с каждым шагом уменьшают, например, в 10 раз. Для данной задачи этот параметр колеблется от 0.04 до единицы (в зависимости от М). Оптимальное значение шага для покоординатного спуска также зависит от значения М: с ростом М шаг изменяется от 0.5.(при М=1) до 2 (при М=14).

Рассмотренные выше способы оптимизации состава и структур высокопроизводительных ВС можно использовать для проектирования ВС (полно- и неполносвязных), обеспечивающих обработку запросов так, чтобы стоимость была не больше заданной, а время ответа было минимальным. Также результаты работы можно использовать для оптимального выбора быстродействия процессоров, количества серверов и клиентов, необходимого количества оборудования, для расчета основных характеристик (4)-(8) для системы клиент - сервер.

Литература

1. Фельдман Л.П., Михайлова Т.В. Методы оптимизации состава и структуры высокопроизводительных систем //Научные труды Донецкого государственного технического университета. Серия «Информатика,кибернетика и вычислительная техника»(ИКВТ-2000).-Донецк:ДонГТУ.-2000
2. Клейнрок Л. Вычислительные системы с очередями. - М.:Мир, 1979, 600с. Последовательно - параллельные вычисления: Пер. с англ. - М.: Мир, 1985. - 456 с.
3. Основы теории вычислительных систем/С.А.Майоров, Г.И.Новиков, Т.И.Алиев и др. М.: Высшая школа, 1978, 408с.
4. Химмельблау Д. Прикладное нелинейное программирование. -М.:Мир, 1975, 535с.
5. Ермольев Ю.М. и др. Математические методы исследования операций. -К.:Вища школа, 1979,312с.