

Д.В. Надеев (ассист.)

Донецкий национальный технический университет
ndv@cs.dgtu.donetsk.ua

ПОДСИСТЕМА БАЛАНСИРОВАНИЯ ЗАГРУЗКИ В ПРОБЛЕМНО-ОРИЕНТИРОВАННОЙ РАСПРЕДЕЛЕННОЙ ПАРАЛЛЕЛЬНОЙ МОДЕЛИРУЮЩЕЙ СРЕДЕ

Описана имплементация методов и средств подсистемы балансирования загрузки в проблемно-ориентированную распределенную параллельную моделирующую среду для шахтных вентиляционных сетей с сосредоточенными параметрами.

проблемно-ориентированная параллельная моделирующая среда, подсистема балансирования загрузки, целевая параллельная вычислительная система, спецификация параллельной модели, динамическая система

Введение

Основной задачей распределенных параллельных моделирующих сред (РПМС) различного назначения [1] является эффективная реализация моделей сложных динамических систем на целевых параллельных вычислительных системах (ЦПВС). Для решения этой задачи в РПМС была выделена подсистема балансирования загрузки [2, 3] ее ресурсов, в рамках подсистемы определена структура модели загрузки РПМС[4], реализованы отдельные ее компоненты [5]. Рассмотрим имплементацию подсистемы балансирования загрузки в РПМС, ориентированной на моделирование шахтных вентиляционных сетей (ШВС) как динамических систем с сосредоточенными параметрами.

Имплементация подсистемы балансирования загрузки в РПМС

Предлагается следующая структурная схема имплементации подсистемы балансирования загрузки в РПМС для моделирования ШВС как систем с сосредоточенными параметрами (рис. 1).

Здесь СПМ-4 – это спецификация параллельной модели сетевого объекта [6], которая включает: файл inndv.dat, каждая строка которого представляет собой набор следующих параметров “*AKJ EKK QI R K H*”, где *AKJ*, *EKK* - начальный и конечный узлы ветви графа, *QI* – номер ветви,

R – аэродинамическое сопротивление, K – аэродинамический коэффициент, определяющий скорость переходного процесса, H – депрессия вентиляторов; файл *tabu.dat*, содержащий начальные значения расходов воздуха в ветвях антидерева сетевого объекта; файл *ru_m.cpp*, параллельная реализация решателя уравнений сетевого динамического объекта с сосредоточенными параметрами в виде масштабируемой MPI-программы со встроенными средствами балансирования загрузки.

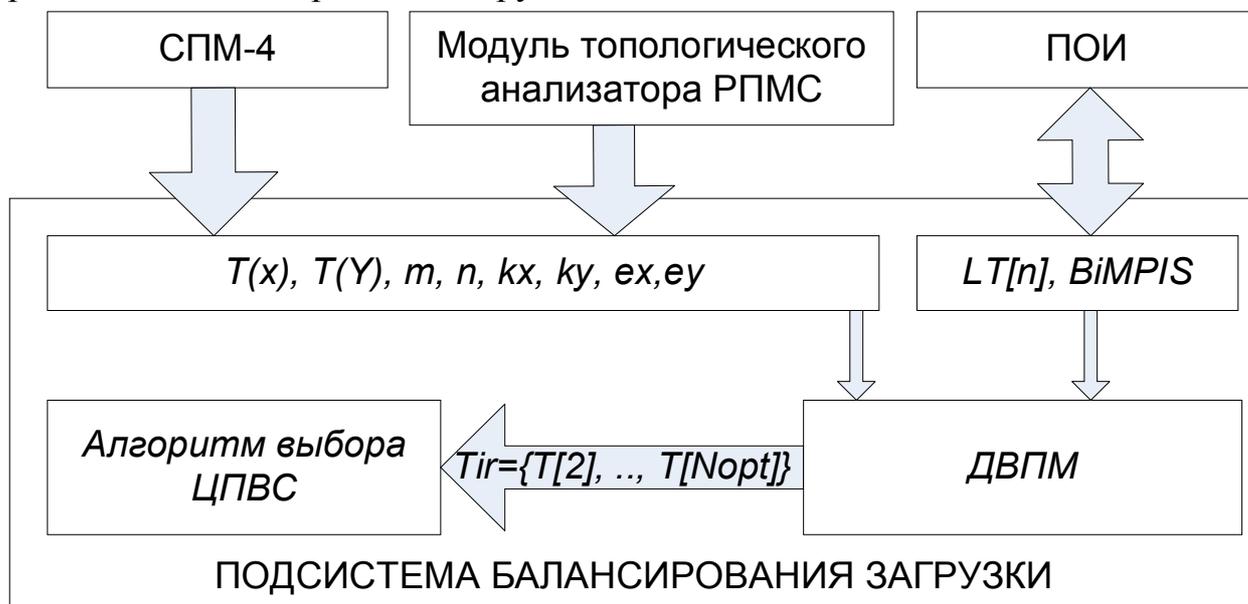


Рис. 1. Структурная схема имплементации ПБЗ в РПМС.

Для принятия решения в подсистему балансирования загрузки передаются следующие параметры:

$T(X)$ – время вычисления виртуального X-процесса для эталонных модели и ЦПВС;

$T(Y)$ – время вычисления виртуального Y-процесса для эталонных модели и ЦПВС;

m – количество виртуальных Y-процессов, – количество ветвей антидерева;

n – количество виртуальных X-процессов, – количество ветвей дерева;

kx – коэффициент изменения времени вычисления виртуального X-процесса по сравнению с эталонной моделью;

ky – коэффициент изменения времени вычисления виртуального Y-процесса по сравнению с эталонной моделью.

После этого выполняется девиртуализация виртуальной параллельной модели (ДВПМ). Первым этапом здесь является оптимизация виртуальной параллельной модели по относительным сложностям виртуальных процессов. Вторым этапом – определение максимального количества процессоров, на которое целесообразно распределять виртуальные процессы по критериям равномерной загрузки и минимизации времени выполнения – N_{max} . Затем вычисляются времена выполнения модели для всех целевых параллельных вычислительных систем, доступных в РПМС, для количества

процессоров от 2 до N_{max} . Так как модели распараллеливаются по MIMD-принципу, то при выполнении условия $T(N+1) > T(N)$ прекращается дальнейшее наращивание N (T – время выполнения модели, N – выбранное количество процессоров для распределения) и принимается N_{opt} равным N . Для расчета времен, затрачиваемых на обмен, используются следующие параметры:

- $LT[N]$ – таблица задержек используемых функций обмена MPI;
- $ViMPIS$ – количество передаваемых данных в функциях обмена, определяющее максимальное время обмена.

Эти параметры поступают в подсистему балансирования загрузки из подсистемы обмена информацией (ПОИ) [1, 2].

Затем формируются векторы времен выполнения модели для всех ЦПВС, входящих в РПМС – T_{ir} и передаются в алгоритм выбора ЦПВС.

Девиртуализация модели в ПБЗ на примере сетевого объекта шахтной вентиляционной сети

Из структуры параллельной MPI-программы решателя уравнений (ru_m.cpp) получается виртуальная параллельная модель, которая состоит из двух типов виртуальных процессов:

- X_n – вычисление расходов воздуха в ветвях дерева, где n – количество ветвей дерева;
- Y_m – вычисление расходов воздуха в ветвях антидерева, где m – количество ветвей антидерева.

Был запущен тестовый объект (эталонный) на эталонной параллельной вычислительной системе VOLVOX со следующими параметрами:

- 567 ветвей в графе;
- 366 узлов;
- 2 процессора. На первом процессоре вычисляются 365 виртуальных процессов X , на втором – 202 виртуальных процессов Y ;
- 70000 – количество итераций;

Получаем количество узлов антидерева:

$$m = 567 - 366 + 1 = 202 \quad (1)$$

Время, затраченное на вычисление виртуальных процессов X и Y , получаем как:

$$T_{cpu}(X) = (C_1 + C_2 + C_3) / D = (38.277443 + 0.221402 + 0.161414) / 365 \approx 0.10487, \quad (2)$$

$$T_{cpu}(Y) = (C_1 + C_2 + C_3) / G = (9.296701 + 0.146100 + 0.664798) / 202 \approx 0.046023, \quad (3)$$

где C_1, C_2, C_3 – экспериментально определенное затраченное время на вычислительные операции.

Вычислим относительные сложности виртуальных процессов:

$$e_x^p = \frac{T_{cpu}(X)}{T_{cpu}(X)} = \frac{0.104870}{0.104870} = 1,000000, \quad (4)$$

$$e_y^p = \frac{T_{cpu}(Y)}{T_{cpu}(X)} = \frac{0.046023}{0.104870} = 0.438858. \quad (5)$$

Теперь, имея времена обработки виртуальных процессов, распределяем виртуальные процессы по процессорам ЦПВС, учитывая невозможность объединения процессов разного типа. В таблице 1 представлены результаты тестового запуска модели на эталонной ЦПВС:

- N – количество используемых процессоров для обработки модели;

Таблица 1. Результаты работы тестовой задачи.

N	N_x	N_y	T_{cpuX}	T_{cpuY}	T_{mpi}	$T(n)$
2	1	1	38,277443	9,296701	6,950018	45,227461
3	2	1	17,185903	8,752855	9,815357	27,00126
4	3	1	9,888807	8,760756	10,040998	19,929805
5	4	1	7,861821	8,902119	13,44384	22,345959
6	5	1	6,387997	8,850708	13,829914	22,680622
7	5	2	6,033161	4,694926	16,302656	22,335817
8	6	2	4,726621	4,946109	16,276789	21,222898
9	7	2	4,364271	4,772072	18,70282	23,474892
10	8	2	4,019701	4,657215	19,345581	24,002796

- N_x – количество процессоров, выделенных под виртуальные процессы X-типа;
- N_y – количество процессоров, выделенных под виртуальные процессы Y-типа;
- T_{cpuX} – время выполнения вычислительных операций виртуальных процессов X-типа на одном процессоре;
- T_{cpuY} – время выполнения вычислительных операций виртуальных процессов Y-типа на одном процессоре;
- T_{mpi} – общее время, затраченное на операции обмена;
- $T(N)$ – общее время обработки задачи.

Время, затраченное на вычислительные операции, вычисляется как:

$$T_{cpuX} = div\left(\frac{365}{N_x}\right)T_{cpu}(X) \cdot r \cdot k_x, \quad (6)$$

$$T_{cpuY} = div\left(\frac{202}{N_y}\right)T_{cpu}(Y) \cdot r \cdot k_y, \quad (7)$$

где r - коэффициент прироста производительности ЦПВС по сравнению с эталонной системой; принимаем единице, так как кластер VOLVOX был выбран в качестве эталонной системы;

k_x, k_y - коэффициенты изменения времени выполнения виртуальных процессов x - и y -типов от количества исходных данных; принимаем единице, так как тестовый объект принят за эталонный.

Время, затраченное на вычислительные операции модели, определяется как максимум $T_{срх}$ и $T_{сру}$.

Расчетные времена выполнения виртуальных процессов тестовой модели приведены в таблице 2.

Таблица 2. Расчетные времена $T_{сру}$.

N_x	N_y	$T_{срх}$	$T_{сру}$	$T_{сру}$
1	1	38,277550	9,296646	38,277550
2	1	19,138775	9,296646	19,138775
3	1	12,759183	9,296646	12,759183
4	1	9,569388	9,296646	9,569388
5	1	7,655510	9,296646	9,296646
5	2	7,655510	4,648323	7,655510
6	2	6,379592	4,648323	6,379592
7	2	5,468221	4,648323	5,468221
8	2	4,784694	4,648323	4,784694

Коэффициенты равномерности загрузки по экспериментальным данным и теоретическим приведены в таблице 3

Таблица 3. Коэффициенты равномерной загрузки ресурсов

$k_{теор.}$	$k_{эксп.}$	Погрешность, %	N_x	N_y
0,621437419	0,621438376	0,000	1	1
0,828583226	0,846923421	2,166	2	1
0,932156129	0,937374896	0,557	3	1
0,994299871	0,906512326	8,829	4	1
0,852891176	0,768124859	9,939	5	1
0,887767742	0,936624788	5,216	5	2
0,932156129	0,966718081	3,575	6	2
0,96668043	0,933534415	3,429	7	2
0,994299871	0,890490089	10,440	8	2

В виртуальном процессе тестового объекта участвует три функции обмена: $MPI_Allreduce$ – 2 функции, $MPI_Allgatherv$ – 1 функция.

Используя тест эффективности основных операций MPI из системы тестов производительности для параллельных компьютеров, по экспериментальным данным строится таблица латентностей LT для кластера VOLVOX – время выполнения функции при обработке сообщения нулевой длины (Таблица 4).

Таблица 4. Таблица латентностей функций обмена MPI для VOLVOX.

N	$MPI_Allreduce$ (мкс)	$MPI_Allgatherv$ (мкс)
2	17.9	17.6
3	37.7	39.5
4	40.7	40.1
5	53.4	60.8
6	60.4	61.3

7	63.2	61.5
8	60.5	60.7
9	71.5	81.7
10	76.5	82.9

Кластер VOLVOX обеспечивает топологию «каждый с каждым» на скорости 1Гбит в дуплексном режиме. Время, потраченное на функции обмена, определяется как:

$$T_{mpii}(N) = (LT_i[N] + B_i^{MPIS} \cdot BR) \cdot LI, \quad (8)$$

где i – идентификатор функции обмена;

$LT_i[N]$ – латентность функции обмена;

BR – пропускная способность системы коммуникаций;

LI – количество итераций;

B_i^{MPIS} – максимальная цепочка передаваемых данных в функции обмена i -ого типа.

Из анализа функций обмена виртуального процесса получаются следующие формулы расчета B_i^{MPIS} :

$$B_1^{MPIS}(N) = 4 \cdot m \cdot (N - 1), \quad (9)$$

$$B_2^{MPIS}(N) = 4 \cdot (N - 1), \quad (10)$$

$$B_3^{MPIS}(N) = 4 \cdot m \cdot N. \quad (11)$$

По формулам (6)-(11) вычисляются времена обменов и общее время выполнения параллельной модели – таблица 5.

Таблица 5. Расчетные времена времени выполнения модели

n	T_{mpi}	T_{pi}	$T(n)$, теор.	$T(n)$	Погрешность, %
2	4,58289584	38,27755	42,860446	45,227461	5,233580
3	9,52208926	19,138775	28,660864	27,00126	5,790489
4	10,61828268	12,759183	23,377466	19,929805	14,747795
5	14,4794761	9,569388	24,048864	22,345959	7,081021
6	16,12866952	9,296646	25,425316	22,680622	10,795121
7	17,16886294	7,65551	24,824373	22,335817	10,024648
8	17,36905636	6,379592	23,748648	21,222898	10,635344
9	21,01324978	5,468221	26,481471	23,474892	11,353519
10	22,4314432	4,784694	27,216137	24,002796	11,806750

Погрешность между теоретическими и практическими результатами (-5%..+15%) объясняется тем, что современные процессорные системы имеют сложные алгоритмы оптимизации, которые зависят от природы обрабатываемых данных.

Данный алгоритм девиртуализации позволяет однозначно определить количество процессоров, на котором модель будет выполняться минимальное время. Получается, что для кластера VOLVOX оптимальное количество процессоров равно трем. Получаем ускорение по сравнению с последовательным алгоритмом – 2,39.

Выводы

Использование программных средств подсистемы балансирования загрузки РПМС приведет к повышению качества параллельного моделирования сложных динамических систем. Девиртуализация ВПМ позволяет комплексно решить проблему оптимизации обработки моделей СДС в РПМС по критериям равномерной загрузки ресурсов и минимизации времени выполнения. Внедрение в РПМС подсистемы балансирования загрузки позволяет оптимально загружать ресурсы целевых вычислительных систем по критерию равномерного распределения ресурсов и позволяет получить минимально возможное время обработки заявки пользователя.

Список литературы

1. Розподілене паралельне моделююче середовище / Надеєв Д., Ротермель К., Святний В.А., Солонін О.М., Степанов І.С., Цайтц М. // Збірник наукових праць ДонНТУ Серія: Проблеми моделювання та автоматизації проектування динамічних систем. - 2001. – Вип. 29. – С. 229-233.
2. Die Subsysteme einer verteilten parallelen Simulationsumgebung für dynamische Systeme / Svjatnyj V.A., Nadeev D.V., Solonin A.N., Zeitz M., Rothermel K. // Збірник наукових праць ДонНТУ Серія: Проблеми моделювання та автоматизації проектування динамічних систем. - 2005. – Вип. 78. – С. 261-268.
3. Святний В.А. Підсистема балансування завантаження ресурсів розподіленого паралельного моделюючого середовища / В.А.Святний, Д.В. Надеєв // Наукові праці Донецького національного технічного університету. Серія: Інформатика, кібернетика та обчислювальна техніка. – 2002. – Вип. 39. – С. 264-270.
4. Надеєв Д.В. Структура модели загрузки распределенной параллельной моделирующей среды. / Д.В. Надеєв // Збірник наукових праць ДонНТУ Серія: Проблеми моделювання та автоматизації проектування динамічних систем. – 2006. – Вип. 78. – С. 44-52.
5. Надеєв Д.В. Девиртуализация виртуальных параллельных моделей сложных динамических систем по критериям балансирования загрузки / Д.В. Надеєв // Наукові праці ДонНТУ. Серія: Проблеми моделювання та автоматизації проектування динамічних систем. - 2008. – Вип. 7(150). – С. 150-157.
6. Смагин А.Н. Моделирование промышленных шахтных вентиляционных сетей / А.Н.Смагин, Т.В.Завадская, Д.В. Надеєв // Наукові праці Донецького національного технічного університету. Серія: Проблеми моделювання та автоматизації проектування динамічних систем. – 2010 – Вип. 8 (168). С. 15-23.

Надійшла до редакції: 05.12.2010

Рецензент: д-р техн.наук, проф. Святний В.А.

Д.В. Надеєв

Донецький національний технічний університет

Подсистема балансирования загрузки в проблемно-ориентированной распределенной параллельной моделирующей среде. Описана имплементация методов и средств подсистемы балансирования загрузки в проблемно-ориентированную распределенную параллельную моделирующую среду для шахтных вентиляционных сетей с сосредоточенными параметрами.

проблемно-ориентированная параллельная моделирующая среда, подсистема балансирования загрузки, целевая параллельная вычислительная система, спецификация параллельной модели, динамическая система

D.V. Nadeev

Donetsk National Technical University

Load Balancing Subsystem in Problem-Oriented Distributed Parallel Simulation Environment. This article describes the implementation methods and tools for subsystem of load balancing in problem-oriented distributed parallel modeling environment for the mine ventilation network with lumped parameters.

problem-oriented distributed parallel environment, subsystem of load balancing, task parallel computing system, specification of parallel model, dynamic system