ИСПОЛЬЗОВАНИЕ ВИРТУАЛЬНЫХ ТОПОЛОГИЙ НА МУЛЬТИКОМПЬЮТЕРАХ

А.В. Черкасов

Новосибирский государственный технический университет

Введение

На сегодняшний день наиболее распространённым средством параллельной реализации алгоритмов численного моделирования является MPI (Message Passing Interface) [3]. MPI представляет собой набор спецификаций для написания параллельных программ (ПП), взаимодействующих по средством обмена сообщениями. Известно, что работа таких программ состоит из двух чередующихся фаз: фазы вычислений и фазы обменов. Следовательно, эффективность работы ПП зависит не только от качественного программирования вычислений в узлах многопроцессорной вычислительной системы (МВС), но и от продуманной организации взаимодействия разных частей программы (подробнее см. [1] и [2]).

В работе рассматривается зависимость времени выполнения межпроцессорных обменов в MPI-программе от использования в ней различных типов виртуальных топологий и различных способов рассылки данных. Исследование проведено на разных мультикомпьютерах. Для проведения эксперимента реализована MPI-программа, которая конструирует заданную виртуальную топологию и измеряет пропускную способность всех связей в ней. Пропускная способность определяется как время передачи данных установленного размера между узлами мультикомпьютера.

1. Среда эксперимента

Исследование проводилось на следующих мультикомпьютерах:

- кластер МВС-1000/М, Москва (768CPUs: 2×Alpha 21264 (667 MHz), Myrinet2000, шесть 128-входовых коммутаторов) [4];
- кластер МВС-1000/М, Новосибирск (32CPUs: 2×Alpha 21264 (833 MHz), Myrinet1000, 16-входовой коммутатор) [5];
- кластер T-Forge32 (32CPUs: 2×Opteron244 (1,8 GHz), Gigabit Ethernet, 16-входовой коммутатор) [6].

2. Использование различных типов ВТ

Под виртуальной топологией (ВТ) будем понимать заданное соединение некоторого множества виртуальных процессоров. Механизм ВТ даёт

разработчику мощный и гибкий инструмент для отображения ПП на физические ресурсы многопроцессорной системы. При этом программная среда автоматически осуществляет отображение виртуальных связей, используемых в программе, на физические коммуникации МВС.

Можно априорно предположить, что в случае, когда все процессоры МВС взаимодействуют через один коммутатор, т.е. по полному графу, то время взаимодействия любой пары процессоров такой системы одинаково с точностью до несущественных отклонений. Следовательно, одна и та же MPI-программа, использующая разные ВТ, размер которых не превосходит числа физических процессоров системы, будет одинаково быстро работать. Это объясняется тем, что любой граф, которым может быть представлена ВТ, является подграфом полного графа, представляющего физические связи кластера. Пример отображения ВТ на некоторую МВС с одним коммутатором приведен на рис.1.

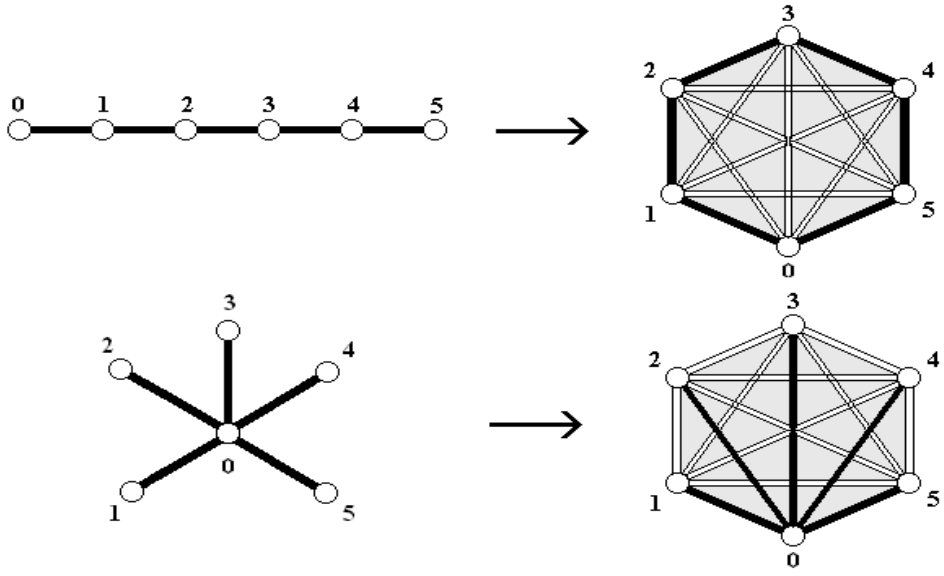

Рис. 1 Отображение виртуальных топологий “линейка” и “звезда” на коммуникации 6-процессорной МВС

В эксперименте исследованы классические типы ВТ, такие как линейка, 2D и 3D решётки, тор, звезда, дерево и полный граф. Размер передаваемого сообщения – 5 Мб. Для определения пропускной способности связи производилось 100 пересылок тестового сообщения. Исследование проводилось только для взаимодействий типа “точка–точка”.

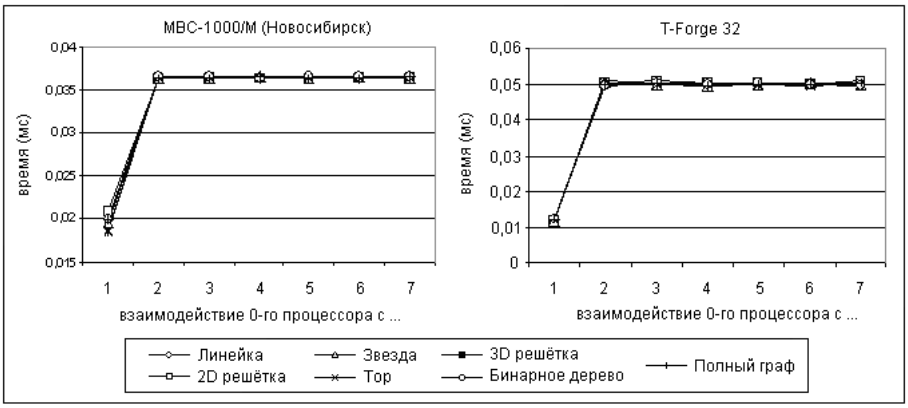

Рис. 2 Зависимость времени передачи сообщений от типа ВТ

Зависимость времени межпроцессорного обмена от типа ВТ для кластеров с одним коммутатором представлена на рис.2. Размер тестовой топологии фиксирован – 8 процессоров. Как видно из рисунка, из общей картины выделяется время взаимодействия 0-го процессора с 1-м процессором. Это объясняется тем, что тестируемые кластеры состоят из двухпроцессорных SMP-узлов и взаимодействие процессоров одного узла происходит быстрее, чем взаимодействие процессоров разных узлов (в приведенном примере 0-й и 1-й процессоры находятся в одном вычислительном узле).

Выводы. Из результатов следует, что наше априорное предположение получило практическое подтверждение. А именно: на кластерах с одним физическим коммутатором нет разницы, каким типом ВТ пользоваться.

Однако для “больших” кластеров, в которых количество коммутаторов больше одного, ситуация меняется: физическая топология системы может оказывать влияние на время работы программы, хотя обычно это ослабляется использованием более быстрых коммуникационных устройств.

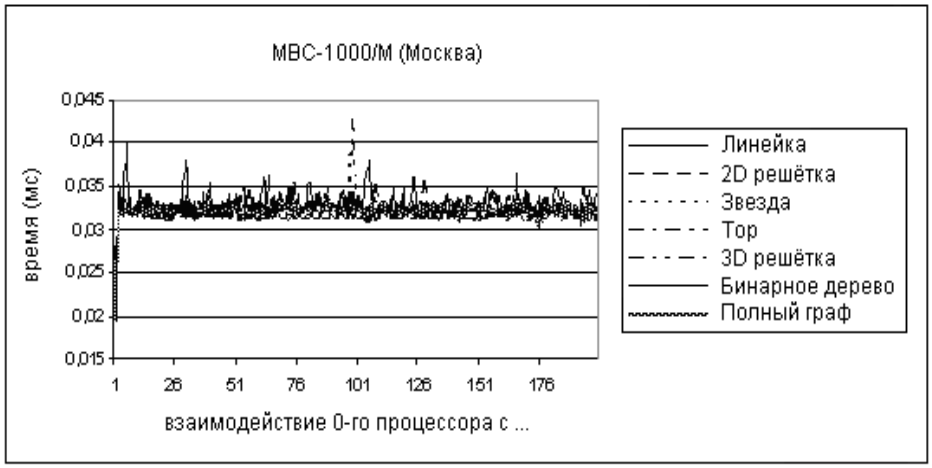

Рис. 3 Зависимость времени передачи сообщений от типа ВТ

Рассмотрим в качестве “большого” кластера московский мультикомпьютер МВС-1000/М. Результаты тестирования этой системы на ВТ размерностью 200 процессоров приведены на рис. 3. В данном случае разброс значений времени для различных ВТ больше, чем, например, для новосибирского кластера. Причина этого заключается в том, что запускаемая программа размещается на процессорах, осуществляющих передачу сообщений более чем через один коммутатор. Поэтому время взаимодействия разных пар процессоров различается.

Выводы. Исходя из результатов, полученных для московского кластера МВС-1000/М, нельзя выделить ВТ какого-либо типа как наиболее предпочтительную для организации межпроцессорных взаимодействий на данном кластере. Все рассмотренные типы ВТ являются одинаково эффективными.

В общем случае для “больших” кластеров рекомендуется провести сначала тестирование кластера на различных ВТ, а затем, основываясь на полученных результатах, выбрать ту или иную ВТ.

3. Способы рассылки данных в MPI-программах

Теперь несколько слов о способах рассылки данных в ПП. Допустим, есть некоторый процессор P, которому нужно разослать свои блоки данных D1, D2,…, DN процессорам P1, P2,…, PK. Это можно сделать двумя основными способами. Первый способ: послать один за другим блоки данных D1,…, DN процессору P1, а затем – процессору P2,…, процессору PK. Второй (ортогональный) способ: сначала послать блок данных D1 процессорам P1, P2,…, PK; затем отправить этим процессорам блок данных D2,…, DN.

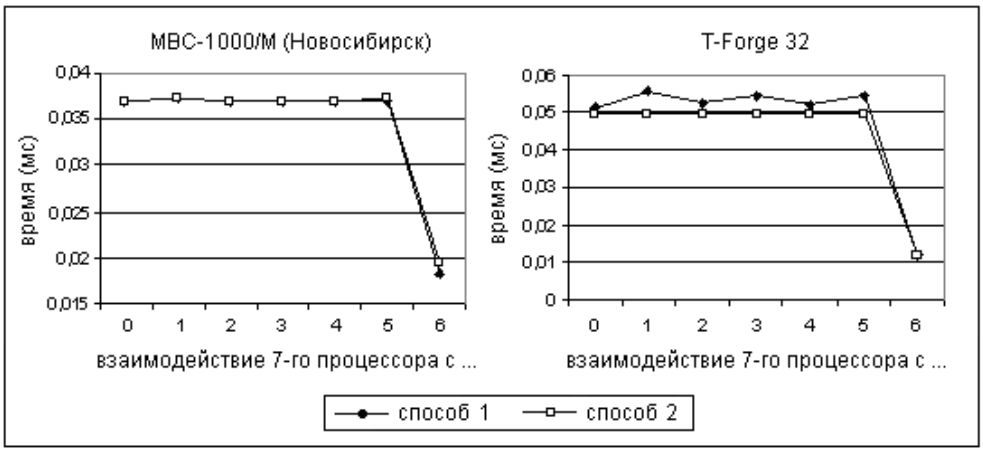

Рис. 4 Зависимость времени передачи данных в ВТ от способа их рассылки

Как видно из рис. 4 и 5, разные способы рассылки сообщений на разных мультикомпьютерах работают не одинаково быстро. Причиной тому могут быть разные реализации MPI, разные размеры системного буфера, а также разное устройство и организация буферов обмена у разных коммуникационных устройств (что имеет место на тестируемых кластерах МВС-1000/М и T-Forge32).

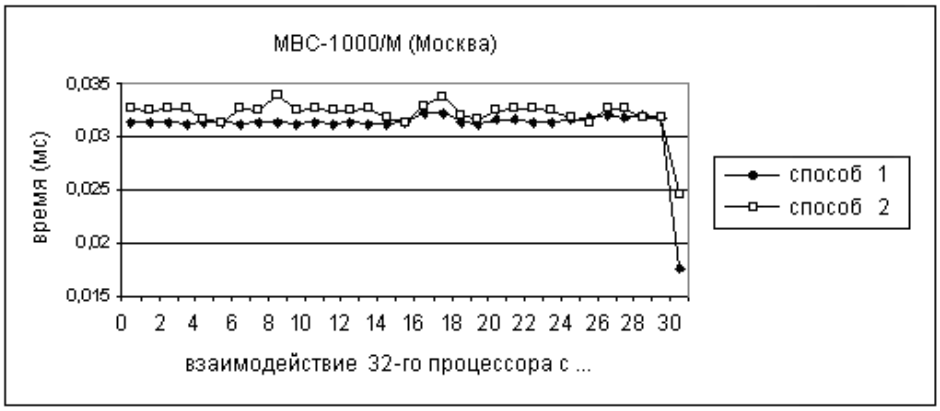

Рис. 5 Зависимость времени передачи данных в ВТ от способа их рассылки

Выводы. При организации межпроцессорного взаимодействия способом 1 на кластере МВС-1000/М (Москва), время выполнения ПП, по крайней мере не меньше, чем способом 2. Для кластера же T-Forge 32, наоборот, второй способ даёт лучшие показатели времени, чем первый. И, наконец, на системе МВС-1000/М (Новосибирск) оба способа работают одинаково быстро. Эти данные можно использовать при написании эффективных ПП для рассмотренного набора мультикомпьютеров.

Литература

1. Малышкин В.Э. Параллельное программирование мультикомпьютеров: Учебное пособие. – Ярославль: Изд-во Ярослав. ун-та, 1999. – 134 с.
2. Малышкин В.Э. Основы параллельных вычислений: Учебное пособие. – Новосибирск: Изд-во НГТУ, 1998. – 60 с.
3. Корнеев В.Д. Параллельное программирование в MPI. – Новосибирск: Изд-во СО РАН, 2000. – 213 с.
4. www.jscc.ru – Межведомственный суперкомпьютерный центр.
5. www2.sscc.ru – Сибирский суперкомпьютерный центр.
6. www.t-platforms.ru/opteron.html – ЦКТ, кластер T-Forge 32.