УДК 004.724.4(045)

Кулаков Ю.А., д-р. техн. наук
Аленин О.И.

СПОСОБ ФОРМИРОВАНИЯ РАСПРЕДЕЛЕННОГО ВИРТУАЛЬНОГО КЛАСТЕРА В СИСТЕМЕ ИЗ НЕСКОЛЬКИХ КЛАСТЕРОВ, СОЕДИНЕННЫХ ЧЕРЕЗ ГЛОБАЛЬНУЮ СЕТЬ

Национальный технический университет Украины "КПИ"

Рассмотрены вопросы, возникающие при объединении нескольких кластерных систем в единый вычислительный ресурс. Сделан обзор способов размещения задач в системах с разной структурой сети между кластерами. Приведено математическое описание стоимости и времени решения задач в системе с несколькими кластерами, соединенными через глобальную сеть. Предложен модифицированный способ размещения задач в такой системе

Введение

В настоящее время наибольшую часть среди высокопроизводительных вычислительных систем составляют системы с кластерной архитектурой [1], это объясняется, прежде всего, их более высоким отношением производительности к стоимости в сравнении с системами других типов. С целью дальнейшего повышения производительности предпринимаются попытки объединить вычислительные ресурсы нескольких систем в один вычислительный ресурс в рамках *grid*-систем. Однако у существующих на сегодняшний день реализаций *grid*-систем есть ряд недостатков: они сложны в установке и настройке, не всякое прикладное программное обеспечение может использоваться в *grid*-среде. Большую часть параллельных прикладных программ для научных и инженерных расчетов предполагалось использовать на кластерных системах. Другой подход заключается в создании распределенного виртуального кластера. Есть ряд подзадач, которые надо решить в процессе его построения:

– создание на основе неоднородных физических однородных виртуальных ресурсов.

– создание системы управления виртуальными и физическими ресурсами.

– объединение узлов каждого виртуального кластера в одну подсеть.

Сравнение типов виртуализации, их достоинства и недостатки описаны в [2].

В [3] произведен анализ влияния таких параметров, как задержка и пропускная способность канала на производительность кластера при решении эталонной задачи. Выбор наиболее подходящих ресурсов для объединения в распределенный виртуальный кластер является нетривиальной задачей, и оптимальность результата зависит как от параметров кластеров и структуры сети между ними, так и от параметров задач.

Обзор существующих решений

Несколько расположенных территориально близко кластеров могут быть соединены между собой высокоскоростной локальной сетью, и формировать вычислительный ресурс большего размера, так называемый мультикластер (рис. 1).

Отличительными особенностями мультикластера являются: простая структура сети между кластерами, наличие одного коммутатора и отсутствие маршрутизаторов. Все узлы всех кластеров фактически находятся в одной сети.

Пропускная способность связи между кластерами ограничивается лишь

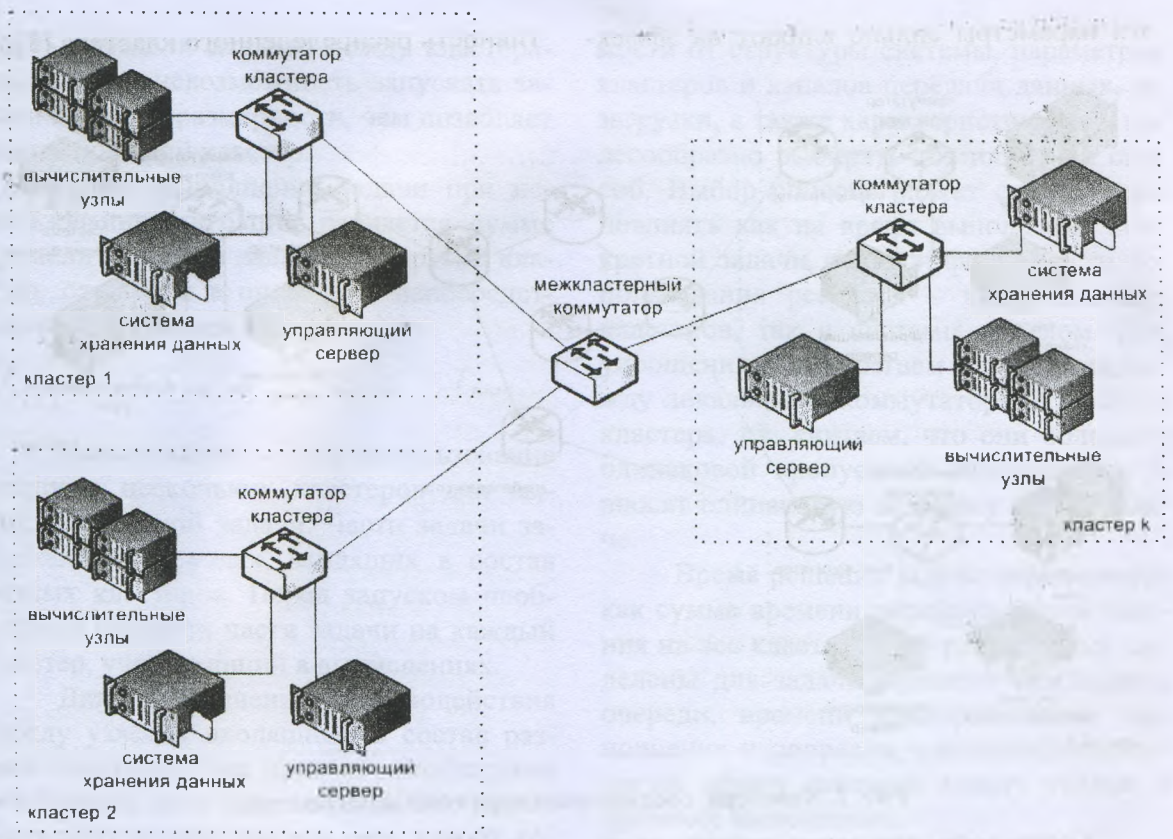

Рис. 1. Мультикластер

пропускной способностью связей кластер-коммутатор.

В работе [4] приведен способ оптимального размещения задач для мультикластера и алгоритм на его основе.

Распределенный суперкомпьютер *ASCI* (*Distributed ASCI Supercomputer*, *DAS*) [5] состоит из пяти кластеров, размещенных в четырех удаленных друг от друга университетах, с суммарным числом узлов около 270, которые объединяются в большую единую систему с помощью специальной оптической сети [6]. Отличительной особенностью *DAS*-3 является новая сеть обмена данными между кластерами, основанная на световых путях (*lightpath*). Используя динамически реконфигурируемые световые пути, приложения могут иметь прямой доступ к управлению сетевой топологией и пропускной способностью. Например, выделяя дополнительный световой путь на некоторой связи, пропускная способность этой связи немедленно увеличивается на 10Гбит/с, без изменений аппаратной инфраструктуры. Когда эта пропускная способность больше не требуется, она может быть передана для использования другой связью или просто освобождена, чтобы в последствии использоваться другим приложением. В исследовательских проектах, таких как *StarPlane, G-Lambda and DRAGON* изучается, как промежуточное программное обеспечение для grid может быть доработано с целью поддержки таких динамически создаваемых световых путей [7].

Постановка задачи

В общем случае кластера соединяются друг с другом через глобальную или городскую сеть (рис. 2). При этом кластера разнообразны по своим характеристикам (количеством вычислительных узлов, их производительностью, объемом памяти и т.д.) и каналам связи. Параметры каналов связи между кластерами могут существенно отличаться по таким характеристикам, как пропускная способность, задержка, потери пакетов. Вместе с тем,

эти параметры сильно влияют на эффективность распределенного кластера [3].

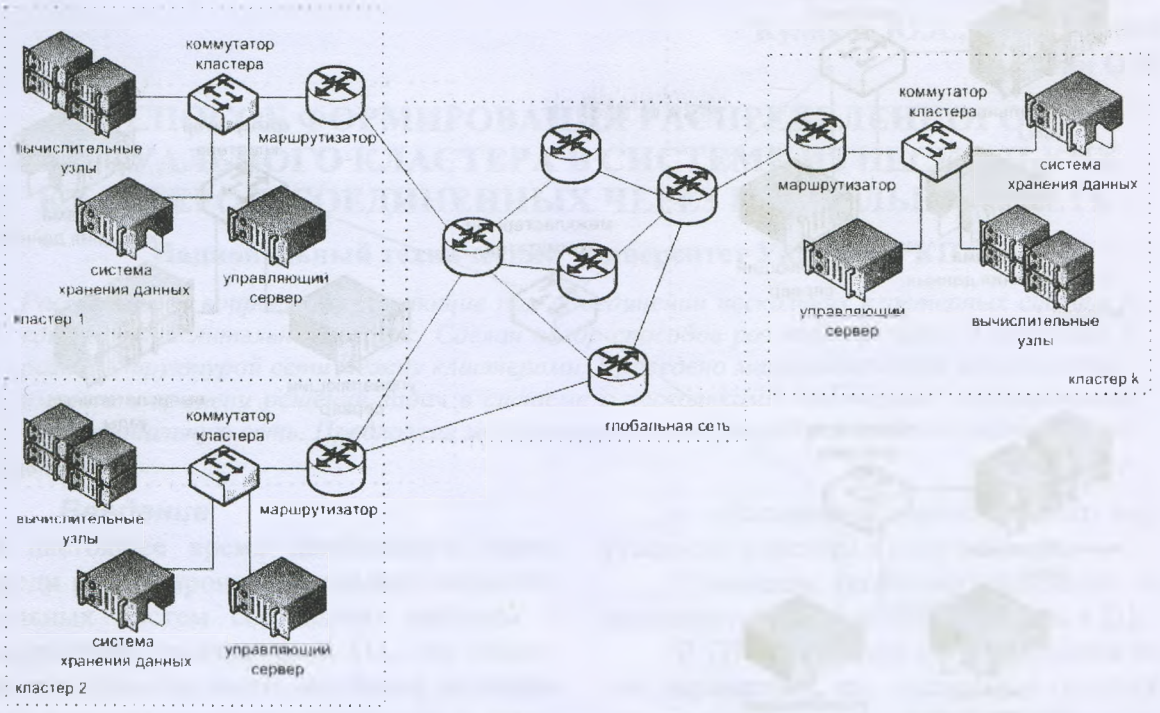

Рис. 2. Кластера, соединенные через глобальную сеть

Кроме того, для задач разного типа оптимальными могут быть разные способы размещения задач.

В зависимости от того, ресурсы каких кластеров выделяются для выполнения задачи, существуют такие способы размещения задач: только локальное размещение, миграция задач и коаллокация [4].

Решение задачи

Локальное размещение подразумевает выделение ресурсов только из того кластера, в очередь которого поступила данная задача. Преимуществами этого способа является простота реализации и отсутствие пересылок данных и задач между кластерами. Недостатками является несбалансированность загрузки кластеров, и невозможность выполнения задач большой размерности. Время выполнения задачи при использовании локального размещения равняется сумме времени нахождения в очереди и непосредственно выполнения задачи.

$$T_{лок.разм.} = T_{вып.задачи} + T_{ожид}$$

Миграция задач – это процесс перемещения задачи с того ресурса, на который оно изначально поступило, на другой ресурс. Задача и связанные с ней файлы данных передаются с локального кластера на другой для выполнения. Обычно решение о миграции задач принимается для увеличения производительности или в связи с вытеснением данной задачи другой, более приоритетной. Один из критических аспектов, определяющих эффективность миграции задач, это стоимость, связанная с передачей файлов данных с локального на удаленный кластер. Кроме того, миграция во время выполнения требует наличия возможности сохранить текущее состояние параллельной программы, включая память и файловые дескрипторы. Это довольно сложная процедура со значительными накладными расходами, сильно зависящая от программы и среды выполнения. Преимуществом данного способа является возможность обеспечить более равномерную загрузку кластеров за счет передачи задач с более загруженных кластеров на менее загруженные. Среди недостатков надо отменить необходи-

мость передачи данных между кластерами, а также невозможность запускать задачи большей размерности, чем позволяет самый большой кластер.

Время выполнения задачи при использовании миграции равняется сумме времени передачи задания на другой кластер, ожидания в очереди и непосредственно выполнения.

$$T_{\substack{\text{миграции} \\ i-j}} = T_{\text{пер .зад .}} + T_{\text{вып .зад .}} + T_{\text{ожид .}}$$

Коаллокация – это использование ресурсов нескольких кластеров для выполнения одной задачи. Части задачи запускаются на узлах, входящих в состав разных кластеров. Перед запуском необходимо передать части задачи на каждый кластер, участвующий в вычислениях.

Для обеспечения взаимодействия между узлами, входящими в состав разных кластеров, как правило, необходимо объединить их в одну подсеть. Это может быть виртуальная частная сеть, или *VLAN*. Преимуществами данного способа является возможность обеспечить равномерную загрузку всех кластеров в системе, а также решить задачу максимальной размерности за минимальное время, использовав при необходимости все узлы всех кластеров.

Недостатками данного способа является необходимость обмена данными между кластерами, (который, как правило, интенсивнее, чем в случае миграции), а также высокая сложность реализации. Кроме того, применение этого способа может существенно повысить время выполнения задачи, части которой интенсивно обмениваются данными. В зависимости от структуры системы, параметров кластеров и каналов передачи данных, их загрузки, а также характеристик задач целесообразно выбирать тот или иной способ. Выбор способа может существенно повлиять как на время выполнения конкретной задачи, так и эффективности использования ресурсов – как отдельных кластеров, так и системы в целом. Для упрощения пренебрегаем различиями между локальными коммутаторами каждого кластера, т.е. считаем, что они обладают одинаковой пропускной способностью и вносят одинаковую задержку при передаче.

Время решения задачи определяется как сумма времени передачи частей задания на все кластера, ресурсы которых выделены для задачи, времени ожидания в очереди, времени непосредственно выполнения и поправки, учитывающей время на обмен данными между узлами в процессе выполнения.

$$T_{\substack{\text{коаллокации} \\ iK}} = T_{\substack{\text{передачи} \\ \text{частей} \\ \text{задачи}}} + T_{\text{ожид.}} + T_{\text{вып.зад.}} + T_{\text{обм.}}$$

Вопросы связанные с выбором ресурсов при использовании способа локального размещения и миграции подробно описаны в литературе. Выбор ресурсов при использовании способа коаллокации для мультикластера описан в [4].Для системы, в которой данные не могут передаваться из одного кластера к нескольким другим одновременно без уменьшения эффективной пропускной способности (например, мультикластер) формулы примут вид:

$$T_{\substack{\text{коаллокации} \\ iK}} = \sum_{k \in K} \frac{D_{data} n_k + D_{OS}}{B_{(i,k)}} + T_{\text{ожид}} + T_{\text{вып задачи}} + \sum_{k \in K} \sum_{l \in K} \frac{n_k n_l bw}{B_{(l,k)}}$$

$$C_{\substack{\text{коаллокации} \\ iK}} = \sum_{k \in K} \frac{D_{data} n_k + D_{OS}}{\bar{B}_{(i,k)}} C_{(i,k)} + T_{\text{ожид}} C_1 + T_{\text{вып зад}} C_2 + \sum_{k \in K} \sum_{l \in K} \frac{n_k n_l bw}{B_{(l,k)}} C_{(l,k)},$$

где C_1- суммарная стоимость использования локальных ресурсов кластеров до запуска задачи, C_2- суммарная стоимость использования локальных ресурсов кластеров во время выполнения задачи. В другом граничном случае, когда между всеми кластерами есть связи, время обмена данными будет равно:

$$T_{обмданными} = \underset{k \in K, l \in K}{MAX} \frac{n_k n_l bw}{B_{(i,k)}}$$

На рис. 3 показана зависимость отношения времени решения задачи на одном узле ко времени решения на n узлах для разной физической топологии сети между кластерами при фиксированных значениях отношения размера блока данных к пропускной способности каналов между кластерами. Предполагается, что обмены данными осуществляются между всеми процессами на всех узлах (каждый с каждым), и независимо от количества узлов, каждый получает весь блок данных. Графики приведены для таких топологий:

1) система с двумя кластерами, выбирается одинаковое число узлов из каждого кластера для объединения в виртуальный кластер;

2) система с двумя кластерами, соотношение количества выбранных узлов 20% и 80%;

3) система с 3-мя кластерами, связи между кластерами по типу каждый-с-каждым, равное количество узлов;

4) система с 3-мя кластерами, межкластерные связи соединяются в одной точке, равное количество узлов;

5) как и в 3, но соотношение узлов 25%, 25% и 50%;

6) как и в 4, но соотношение узлов 25%, 25% и 50%;

7) система с 5-ю кластерами, связи между кластерами по типу каждый-с-каждым, равное количество узлов;

8) система с 5-ю кластерами межкластерные связи соединяются в одной точке, равное количество узлов.

На рис. 4 приведены графики для тех же топологий, но блок данных делится на количество узлов

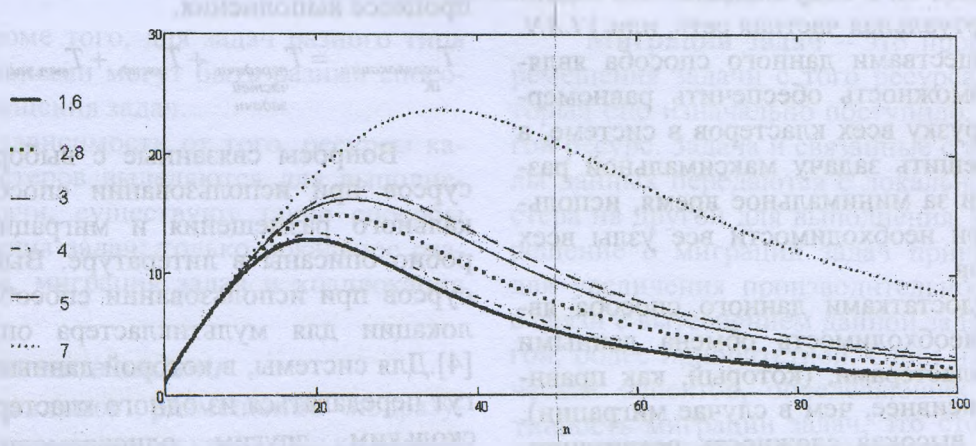

Рис. 3. Фиксированный размер блока

Рис. 4. Переменный размер блока данных

Исходя из формул и графиков, целесообразно применять способы размещения задач в таком порядке: локальное размещение, миграция, коаллокация. Если решение не удовлетворяет ограничениям задачи к минимальному количеству ресурсов (число процессоров, объем памяти), то переходим к следующему способу. Максимальное количество узлов, которые могут быть использованы при построении виртуального кластера, ограничивается максимально допустимой стоимостью решения. Среди решений, находящихся между этими значениями по количеству узлов, выбирается решение с минимальной стоимостью. Если подходящее решения не найдено, задача ожидает освобождения ресурсов.

Выводы

1. Для задач с интенсивным обменом данными между ее частями предпочтительнее использовать локальное размещение или миграцию. Для задач, требовательных к числу процессоров и объему памяти, но менее интенсивным обменом данными, можно использовать любой способ размещения, учитывая ограничения на количество процессоров и объем памяти.

2. При использовании коаллокации оптимальное решение при выборе кластеров следует искать среди таких:

– один большой кластер и несколько маленьких;

– наиболее связанные между собой кластера.

3. Для каждого класса задач существует некоторая величина отношения свободной пропускной способности связей, соединяющих данный кластер с другими, к числу свободных узлов в данном кластере, определяющая эффективность использования этого кластера для построения распределенного виртуального кластера для решения этой задачи. Знание этих величин заранее может позволить более эффективно распределять ресурсы, но это требует дальнейших исследований.

Список литературы

1. TOP500 list of supercomputers [Online source] – 2008. – Link: http://www.top500.org/lists/2008/11

2. *Jones T.* An overview of virtualization methods, architectures, and implementations [Online source]/ T.Jones // IBM - 2006,- Link:http://www.ibm.com/developerworks/linux/library/l-linuxvirt

3. *Аленин О.И.* Анализ влияния сети передачи данных на производительность виртуального распределенного кластера / Проблеми інформатизації та управління: Зб. наук. пр. – К.: НАУ. – 2009. – Вип. 1(25). – С. 15–20.

4. *Jones W.* Improving parallel job scheduling performance in multi-clusters through selective job co-allocation. / PhD thesis.- Clemson University. – 2005.

5. DAS-3 Supercomputer [Online source] – 2008. – Link: http://www.cs.vu.nl/das3/

6. *Grosso P.* StarPlane – an Application Controlled Photonic Network / P. Grosso, J.-P. Velders, L. Xu, C. de Laat // 17th eChallenges Conference. – 24-26 October 2007, Hague, Netherlands. – P. 1476–1481.

7. *Maassen J.* "Assessing the Impact of Future Reconfigurable Optical Networks on Application Performance" / J. Maassen, K. Verstoep, H. E. Bal, P. Grosso and C. de Laat // Sixth High-Performance Grid Computing Workshop – IPDPS 2009, May 25-29, – 2009, Rome, Italy.