УДК 004.75

Альбертьян А.М.[1], Курочкин И.И.[2]

[1] Федеральный исследовательский центр «Информатика и управление» РАН, г. Москва, Россия
[2] Институт проблем передачи информации им. А.А. Харкевича РАН, г. Москва, Россия

ИСПОЛЬЗОВАНИЕ СОПРОЦЕССОРОВ INTEL XEON PHI В ГРИД-СИСТЕМАХ ИЗ ПЕРСОНАЛЬНЫХ КОМПЬЮТЕРОВ*

Аннотация

Использование грид-систем из персональных компьютеров один из распространенных способов организации распределенных вычислений. Была взята самая популярная платформа BOINC для организации грид-системы из персональных компьютеров. Предлагается использовать на вычислительных узлах такой грид-системы не только универсальные процессоры и видеокарты, но и специализированные вычислители Intel Xeon Phi. Использование специализированных вычислителей имеет определенные особенности, которые были продемонстрированы в рамках численного эксперимента на распределенной системе. Для проведения численного эксперимента был использован развернутый проект добровольных распределенных вычислений. В качестве прикладной задачи для численного эксперимента была взята задача поиска диагональных латинских квадратов порядка 9. Наличие большого количества искомых квадратов позволяет использовать эту задачу в качестве тестовой для оценки эффективности различных вычислителей. Обсуждаются оценки производительности Xeon Phi для адаптированного MPI приложения. Показана зависимость количества сгенерированных диагональных латинских квадратов в секунду от количества используемых логических ядер. Проведено сравнение производительности специализированного вычислителя Intel Xeon Phi 5110p с универсальным процессором Intel Xeon E5-1650 ...

Ключевые слова

Распределенные вычисления; грид-системы из персональных компьютеров; BOINC; Intel Xeon Phi; диагональные латинские квадраты.

Albertian A.M.[1], Kurochkin I.I.[2]

[1] Federal Research Center Computer Science and Control of the Russian Academy of Sciences, Moscow, Russia
[2] Kharkevich Institute for information transmission problems of the Russian Academy of Sciences, Moscow, Russia

THE USE OF INTEL XEON PHI COMPRESSOR IN THE DESKTOP GRID SYSTEMS

Abstract

Using desktop grid system is one of the most common ways to organize distributed computing. The most popular BOINC platform was used to organize a desktop grid system. It is suggested to use not only universal processors and video cards on computing nodes of such a grid system, but also specialized Intel Xeon Phi coprocessors. The use of specialized coprocessors has certain features that have been demonstrated in the framework of a numerical experiment on a distributed system. The active project of voluntary distributed computing was used to conduct the numerical experiment. As an applied problem for a numerical experiment, the problem of searching for diagonal Latin squares of order 9 was taken. The presence of a large number of desired squares allows us to use this problem as a test for evaluating the efficiency of various coprocessors. The Xeon Phi performance estimates for the adapted MPI application are discussed. The dependence of the number of generated diagonal Latin squares per second on the number of logical cores used is shown. The performance of the

* **Труды II Международной научной конференции «Конвергентные когнитивно-информационные технологии» (Convergent'2017), Москва, 24-26 ноября, 2017**

Proceedings of the II International scientific conference "Convergent cognitive information technologies" (Convergent'2017), Moscow, Russia, November 24-26, 2017

specialized Intel Xeon Phi 5110p coprocessor with the universal Intel Xeon processor E5-1650 is compared….

Keywords

Distributed computing; desktop grid; BOINC; Intel Xeon Phi; diagonal Latin squares.

Введение

В рамках распределенных вычислений, представляющих собой способ решения трудоемких вычислительных задач с использованием компьютеров, объединенных в вычислительную систему, особый интерес представляют грид-системы из персональных компьютеров(ГСПК) и добровольные распределенные вычисления (volunteer computing). Добровольными распределенными вычислениями называются вычисления с использованием добровольно предоставленных вычислительных ресурсов, организованных в ГСПК.

Существует несколько платформ для организации распределенных вычислений: Globus [1], HTCondor [2], Legion, но самой распространенной на текущий момент является BOINC [3] [4].

На базе платформы BOINC развернуто около сотни публичных международных проектов добровольных распределенных вычислений, к которым подключены около 16 миллионов компьютеров по всему миру [4]. Платформа BOINC имеет архитектуру клиент-сервер.

Подавляющее большинство проектов добровольных распределенных вычислений – научные проекты ведущих мировых университетов и научных организаций. Один проект добровольных распределенных вычислений может выполнять один или несколько экспериментов. Эксперименты в проекте объединяет общая тематика, кроме того, они проводятся одной научной группой.

Примерами действующих международных зонтичных проектов могут служить World Community Grid, LHC@home и Einstein@home. На текущий момент, именно они являются одними из крупнейших проектов по количеству активных вычислительных узлов [4].

Как работает BOINC-проект

Большинство проектов добровольных вычислений организуются для решения одной научной задачи. Для решения задачи требуется проведение серии численных экспериментов. Задача, как правило, может разбиваться на множество независимых подзадач [5]. Каждая подзадача будет рассчитываться на отдельном вычислительном узле распределенной системы. У большинства проектов единственное приложение для проведения одного или серии вычислительных экспериментов. Для различных подзадач используются различные наборы входных данных. Такой тип задач в литературе называется «bag of tasks» [6] или задача, разделяемая по данным. В качестве примера таких задач можно привести задачи комбинаторики [7] и полного перебора, SAT-задачи [8], некоторые задачи машинного обучения, задачи имитационного математического моделирования [9] и др.

В большинстве международных проектов добровольных распределенных вычислений на платформе BOINC расчеты проводятся, либо с использованием центрального процессора, либо с возможностью задействовать вычислительные мощности установленного в системе видеоадаптера AMD или NVIDIA через один из API: OpenCL или CUDA.

Сопроцессор Intel Xeon Phi

Сопроцессоры Xeon Phi являются относительно новым решением фирмы Intel для ускорения расчетов различных научных и инженерных задач, особенностью которых является высокая степень распараллеливания. Данные сопроцессоры являются вариантом реализации архитектуры Intel MIC (Intel Many Integrated Core Architecture), которая предполагает использование множества (от нескольких десятков) универсальных процессоров с архитектурой x86 в составе одного вычислительного устройства с общей памятью. В качестве дополнительных особенностей в этой архитектуре предполагается использование специальных расширенных наборов команд SIMD (Single Instruction – Multiple Data) для работы с векторными операндами размером до 512 бит. В рассматриваемом решении с архитектурой под названием Knight Corner (KNC), используется специальное расширение SIMD MIC, а в новых, только что появившихся на рынке решениях с архитектурой Knight Landing (KNL) – набор команд SIMD AVX-512, совместимый (за некоторыми исключениями) с набором команд AVX-512 для центральных процессоров Xeon поколения SkyLake и будущих поколений (Cannonlake и др.). Сопроцессоры Intel Xeon Phi реализованы в качестве стандартной карты расширения с интерфейсом PCI Express x16 Gen2 (с несколькими вариантами используемого конструктива) и установленными непосредственно на плате 8 ГБ или 16 ГБ оперативной памяти, в вариантах с активным или пассивным охлаждением. В новом поколении (со)процессоров Intel Xeon Phi KNL, предусмотрен вариант их использования в качестве центрального процессора вычислительной платформы, что позволяет непосредственно использовать

многократно больший объем оперативной памяти, а также избавляет от ограничений, накладываемых подключением к системе по шине PCIe.

Изначально архитектура MIC разрабатывалась компанией Intel под названием Larrabee, как решение для ускорения расчетов, связанных с построением и отображением двумерных и трехмерных графических изображений, для замены используемого графического процессора Intel GMA. Также, в современных вариантах архитектуры Intel MIC, используются наработки из проектов многоядерных процессоров Intel Teraflops Research Chip и Intel Single-chip Cloud Computer. Наиболее существенным отличием этого решения от решений, предлагаемых производителями графических процессоров – NVIDIA и AMD (ATI), является использование в качестве вычислительных ядер универсальных процессоров со стандартной архитектурой. В результате этого существенно упрощается их использование, а также адаптация существующего программного обеспечения для научных и инженерных расчетов. Кроме того, для эффективного решения многих задач желательно использовать именно процессоры с полноценным набором инструкций, включающим комплексный набор управляющих команд. И, в конечном итоге, было принято решение использовать данную архитектуру – многоядерный универсальный процессор, в качестве ускорителя для решения вычислительных задач.

Подобные ускорители используются на текущий момент во множестве высокопроизводительных вычислительных систем по всему миру, начиная от небольших вычислительных кластеров, до мощных суперкомпьютерных решений, как, например, входящий в TOP-500 суперкомпьютер Tianhe-2. Данный суперкомпьютер занимал первое место в мире по производительности с июня 2013 по ноябрь 2015 года, и покинул его, исключительно благодаря ограничениям на поставку обновленной элементной базы. Однако, в качестве вычислителей для распределенных вычислений, эти сопроцессоры практически не использовались.

Тестовый стенд

В качестве тестового стенда были использованы 2 сопроцессора Intel Xeon Phi 5110P, установленных в платформу с одним процессором Intel Xeon E5-1650 и 64 ГБ оперативной памяти. В процессе построения вычислительной системы были решены несколько интересных задач инженерного характера, а также проведен выбор относительно доступной и недорогой, но при этом надежной элементной базы, обеспечивающей полноценное функционирование сопроцессоров в системе и возможность работы в режиме 24x7. Для эффективного использования сопроцессоров, необходимо наличие двух дополнительных интерфейсов PCIe x16 работающих на полной скорости, при установленных в эту же систему двух графических адаптерах NVIDIA GTX Titan. А для системы в целом, желательна возможность установки значительного объема оперативной памяти, дополнительного кэширующего контроллера SAS/SATA RAID с интерфейсом PCIe x8 для подсистемы хранения данных и т.д.

Так как данный стенд предполагалось использовать в качестве рабочей станции, то присутствовали существенные ограничения по максимальному уровню шума. В результате, для обеспечения эффективного охлаждения сопроцессоров (изначально созданных с пассивной системой охлаждения и рассчитанных на установку в корпус серверного класса с мощной, но шумной собственной системой охлаждения), было принято решение по их существенной доработке и последующей установке на них системы жидкостного охлаждения (СЖО). В качестве материнской платы была выбрана относительно недорогая материнская плата для рабочих станций – ASUS P9X79-E WS с чипсетом Intel X79 Express, для установки процессора с конструктивом FCLGA2011. На плате установлен процессор Intel Xeon E5-1650 (также с использованием СЖО), обеспечивающий собственные 40 линий интерфейса PCIe и поддержку 64 ГБ оперативной памяти DDR3 с коррекцией ошибок (ECC) в четырехканальном режиме. Особенностью данной материнской платы является интересное решение с использованием двух коммутаторов линий PCIe (PLX Broadcom/Avago PEX8747) и двух дополнительных коммутаторов Quick Switch (QS) с поддержкой PCIe Gen3, которые вместе обеспечивают подключение до четырех плат расширения с интерфейсами PCIe x16 Gen3, функционирующих на полной скорости. Также было осуществлено подключение SAS/SATA RAID контроллера Intel RS2BL080 (LSI 2108) с интерфейсом PCIe x8 Gen2 через удлинитель для слота PCIe x16, установленный в слот расширения материнской платы, который подключен непосредственно к самому процессору, минуя коммутаторы.

Данный тестовый стенд был включен в тестовый проект распределенных вычислений на плаформе BOINC для решения задач поиска пар ортогональных диагональных латинских квадратов (ДЛК) [10].

Вычислительное приложение

Приложение было реализовано на C++ (с использованием возможностей версии C++11, ISO/IEC 14882:2011) как переносимое Windows/Linux MPI приложение, использующее для своей работы Intel MPI Library и Intel Manycore Platform Software Stack (Intel MPSS). При разработке и компиляции приложения были использованы Intel Parallel Studio XE 2017, в первую очередь это библиотека MPI и оптимизирующий

компилятор C++ для архитектур Intel 64, Intel IA-32 и Intel Many Integrated Core (Intel MIC – архитектура сопроцессоров Intel Xeon Phi), а также Microsoft Visual Studio 2015 Update 3. Производительность приложения исследовалась под управлением операционных систем Microsoft Windows 10 Pro и Microsoft Windows Server 2012 R2.

Вычислительное приложение предназначено для перечисления диагональных латинских квадратов (ДЛК, англ. DLS) порядка 9 [10]. В качестве рабочего набора данных для приложения используется текстовый файл, в каждой строке которого приведены заранее сгенерированные жестко детерминированные значения ключевых элементов матрицы размерностью 9. Эти наборы значений используются для построения всех возможных комбинаций ДЛК порядка 9. Задача генерации ДЛК на основе наборов ключевых элементов может быть разделена на множество автономных подзадач: поиск ДЛК на основе одного набора – одна подзадача. Каждая подзадача выполняется в отдельном MPI-процессе.

В процессе работы приложения, для каждой строки рабочего файла, непосредственно после приведенных исходных данных, сохраняется количество найденных в процессе перечисления ДЛК, а также общее время, затраченное на поиск всех ДЛК для этих исходных данных.

Оценка производительности

Были использованы несколько алгоритмов измерения производительности, с различными методиками оценки суммарной производительности для всех вычислительных процессов. Все используемые для оценки производительности алгоритмы основаны на подсчете количества найденных ДЛК для каждого из MPI процессов. Выполняемые в процессе измерения производительности вычислительные процессы, каждые 100M обнаруженных ДЛК, передают управляющему процессу следующую статистику: номер итерации (счетчик 100M квадратов, найденных данным процессом), общее количество найденных данным процессом квадратов и общее затраченное время на поиск в наносекундах (нс).

Исходный алгоритм измерения производительности основывается на подсчете количества возвращенных номеров итераций, отличных от первоначального. Итерация считается завершенной, при достижении значения, равного общему количеству вычислительных процессов. После чего следующий принятый номер итерации используется в качестве первоначального. Необходимое для завершения измерения производительности количество итераций задается в качестве параметра приложения. Были использованы значения 2 для центрального процессора системы и 1 для сопроцессора Intel Xeon Phi. В данном случае, 2 означает ожидание получения двух номеров итерации, отличных от первоначального, то есть около 4 итераций (100 млн. найденных ДЛК) для каждого из вычислительных процессов.

Модифицированный алгоритм измерения производительности был реализован на основе исходного алгоритма, но в нем при подсчете количества завершенных итераций, учитываются только номера итераций, которые больше, чем первоначальный (с учетом возможного переполнения). То есть, при наличии отклонений в производительности различных вычислительных процессов, учет количества завершенных итераций ведется только для последующих номеров итераций, что позволяет уменьшить нестабильность результатов измерения производительности между несколькими запусками приложения. При достижении значения, равного общему количеству вычислительных процессов, первоначальный номер итерации увеличивается на 2 и подсчет начинается снова. Таким образом, не происходит заметного влияния наименее производительных вычислительных процессов на измеренную интегральную производительности приложения.

Эксперимент и результаты

Приложение для генерации диагональных латинских квадратов было оптимизировано для компиляции с помощью Intel Parallel Studio XE с использованием возможностей Intel MPI Library. В результате проведенных вычислительных экспериментов, были получены сравнительные результаты производительности данного приложения на центральном процессоре Intel Xeon E5-1650 и сопроцессорах Intel Xeon Phi 5110P (Таблица 1).

Таблица 1. Сравнительные результаты производительности E5-1650 и 5110P

	Среднее время на квадрат, нс	Кол-во квадратов в секунду
Phi 5110P #0 (1 поток)	2 203	453 729
Phi 5110P #0 (240 потоков)	11	88 819 961
Phi 5110P #1 (1 поток)	2 272	440 016
Phi 5110P #1 (240 потоков)	11	87 598 813
E5-1650 (1 поток)	482	2 073 384

E5-1650 (24 потока)	73	13 587 625
E5-1650 (1 поток)	220	4 535 049
E5-1650 (12 потоков)	22	45 178 456

Термин «поток» при оценке производительности используется в связи с тем, что сопроцессор Intel Xeon Phi 5110P при оценке производительности оперирует именно этим понятием. Следует заметить, что для одного MPI-процесса выделяется одно логическое ядро на процессоре или сопроцессоре.

В результате применения модифицированного метода измерения производительности были получены результаты (Рисунок 1).

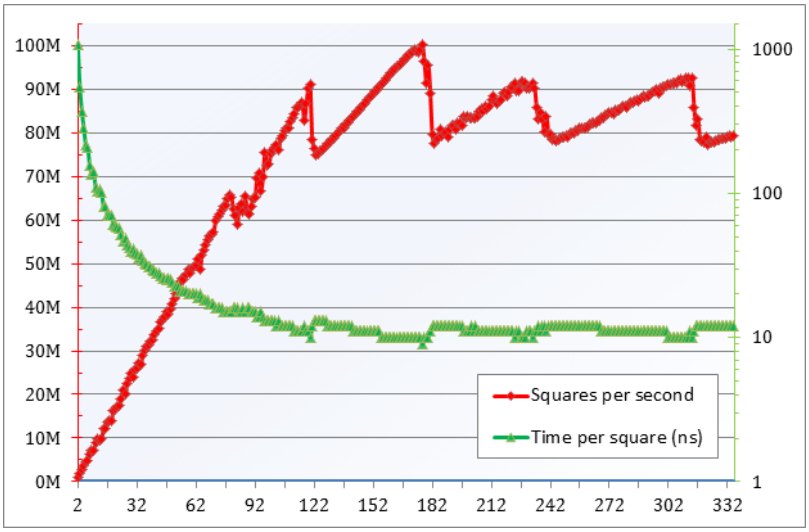

Рисунок 1. Производительность Phi 5110P

После анализа результатов можно сформулировать следующие утверждения:

- Производительность однопоточных вычислений для данного MPI приложения, при исполнении на сопроцессоре Intel Xeon Phi Coprocessor 5110P, практически на порядок (около 6–7 раз) ниже, чем производительность при исполнении на центральном процессоре Intel Xeon Processor E5-1650.
- При этом максимальная полученная суммарная многопоточная производительность для сопроцессора примерно в два раза выше производительности центрального процессора.
- При выборе оптимального количества задействованных MPI процессов следует провести предварительное тестирование для каждого конкретного случая и используемого алгоритма.
- Например, несмотря на то, что производитель рекомендует для сопроцессоров Intel Xeon Phi x100 исходить из расчета исполнения 4 потоков на одно ядро (в Intel Xeon Phi Coprocessor 5110P – 60 ядер, то есть всего 240 потоков), получены ярко выраженные максимумы производительности в районе 119–120, 173–177, 223–234 и т.д. MPI процессов (включая управляющий). Причем абсолютный максимум производительности достигается при 175–177 MPI процессах (то есть при исполнении 3 потоков на ядро, подробности далее).
- Увеличение количества процессов хотя бы на один от зоны максимальной производительности, приводит к резкому падению производительности и постепенному плавному росту при дальнейшем увеличении (до следующего максимума).
- В документации производителя сказано, что одно ядро сопроцессора целиком отводится для исполнения собственного кода операционной системы, обслуживания прерываний и т.д. Полученные результаты показывают, что как минимум еще некоторое количество вычислительных ресурсов используется системой для целей управления выполнением MPI приложения. Это хорошо заметно по смещению, в сторону уменьшения количества потоков, областей максимальной производительности при увеличении количества MPI процессов.
- При этом при достаточном количестве вычислительных ресурсов (когда количество MPI потоков не более, чем 3 · (<MIC Cores> – 1), максимумы производительности практически точно вычисляются по формуле:

$$MPI\ Ranks \approx N(<MIC\ Cores> - 1), \text{ при } N \le 4.$$

Заключение

Продемонстрированная сопроцессором Intel Xeon Phi производительность при многопоточных

параллельных вычислениях показала значительное (в несколько раз) преимущество над центральным процессором, несмотря на более высокую (от 5-10 раз) производительность центрального процессора в однопоточном режиме. Существуют определенные особенности, которые следует учитывать при разработке вычислительных приложений для Intel Xeon Phi. Таким образом, можно утверждать о широких перспективах применения сопроцессоров Intel Xeon Phi при решении задач в грид-системах из персональных компьютеров в рамках проектов распределенных вычислений не только на платформе BOINC, но и на других грид-системах.

Благодарности

Авторы выражают благодарность Заикину О.С. и Ватутину Э.И. за консультации по алгоритмам генерации диагональных латинских квадратов.

Работа поддержана грантом РНФ (№16-11-10352).

Литература

1. I Foster, C Kesselman "Globus: A metacomputing infrastructure toolkit"// International Journal of High Performance Computing Applications 11 (2), 1997, pp.115-128.
2. M.J. Litzkow, M. Livny, M.W. Mutka "Condor-a hunter of idle workstations" // Distributed Computing Systems, IEEE,1988.
3. D.P. Anderson "BOINC: a system for public-resource computing and storage" // Grid Computing, IEEE, 2004
4. . The server of statistics of voluntary distributed computing projects on the BOINC platform. [электронный ресурс] // URL: http://boincstats.com (дата обращения 23.04.2017).
5. Benoit, et al., Scheduling Concurrent Bag-of-Tasks Applications on Heterogeneous Platforms // IEEE Trans. Computers, vol. 59, no. 2, 2010. pp. 202-217.
6. Choi S. J. et al. Characterizing and classifying desktop grid // Cluster Computing and the Grid, 2007. CCGRID 2007. Seventh IEEE International Symposium on. – IEEE, 2007. – pp. 743-748
7. Vatutin, E. I., Valyaev, S. Y., & Titov, V. S. (2015). Comparison of Sequential Methods for Getting Separations of Parallel Logic Control Algorithms Using Volunteer Computing. // BOINC:FAST-2015.
8. Posypkin M., SemenovA. Zaikin O. Using BOINC desktop grid to solve large scale SAT problems. // Computer Science, 13 (1), 2012. pp. 25-34.
9. Kurochkin I., Grinberg Ya., Different Criteria of Dynamic Routing // Procedia Computer Science, Volume 66, 2015, pp 166-173
10. Zaikin O. and Kochemazov S.. The Search for Systems of Diagonal Latin Squares Using the SAT@home Project // International Journal of Open Information Technologies. Vol. 3, No. 11 (2015). pp. 4-9.

References

1. I Foster, C Kesselman "Globus: A metacomputing infrastructure toolkit"// International Journal of High Performance Computing Applications 11 (2), 1997, pp.115-128.
2. M.J. Litzkow, M. Livny, M.W. Mutka "Condor-a hunter of idle workstations" // Distributed Computing Systems, IEEE,1988.
3. D.P. Anderson "BOINC: a system for public-resource computing and storage" // Grid Computing, IEEE, 2004
4. . The server of statistics of voluntary distributed computing projects on the BOINC platform. [электронный ресурс] // URL: http://boincstats.com (дата обращения 23.04.2017).
5. Benoit, et al., Scheduling Concurrent Bag-of-Tasks Applications on Heterogeneous Platforms // IEEE Trans. Computers, vol. 59, no. 2, 2010. pp. 202-217.
6. Choi S. J. et al. Characterizing and classifying desktop grid // Cluster Computing and the Grid, 2007. CCGRID 2007. Seventh IEEE International Symposium on. – IEEE, 2007. – pp. 743-748
7. Vatutin, E. I., Valyaev, S. Y., & Titov, V. S. (2015). Comparison of Sequential Methods for Getting Separations of Parallel Logic Control Algorithms Using Volunteer Computing. // BOINC:FAST-2015.
8. Posypkin M., SemenovA. Zaikin O. Using BOINC desktop grid to solve large scale SAT problems. // Computer Science, 13 (1), 2012. pp. 25-34.
9. Kurochkin I., Grinberg Ya., Different Criteria of Dynamic Routing // Procedia Computer Science, Volume 66, 2015, pp 166-173
10. Zaikin O. and Kochemazov S. The Search for Systems of Diagonal Latin Squares Using the SAT@home Project // International Journal of Open Information Technologies. Vol. 3, No. 11 (2015). pp. 4-9.

Об авторах:

Альбертьян Александр Михайлович, ведущий инженер, Федеральный исследовательский центр «Информатика и управление» Российской академии наук, admin@isa.ru

Курочкин Илья Ильич, кандидат технических наук, старший научный сотрудник лаборатории Ц-1, Институт проблем передачи информации им. А.А. Харкевича Российской академии наук, kurochkin@iitp.ru

Note on the authors:

Albertian Alexander M., engineer, Federal Research Center Computer Science and Control of the Russian Academy of Sciences, admin@isa.ru

Kurochkin Ilya I., Candidate of Engineering Sciences, senior researcher of the laboratory C-1, Kharkevich Institute for information transmission problems of Russian Academy of Sciences, kurochkin@iitp.ru