

15. Blumenthal D.J., Granstrand P., Thylen L. BER floors due to heterodyne coherent crosstalk in space photonic switches for WDM networks. *IEEE Photonic Technology Letters*, 1996, vol. 8, no. 2, pp. 284–286. doi: 10.1109/68.484268.

Received 13.03.2017

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.7; 621.39

ОЦЕНКА МАСШТАБИРУЕМОСТИ ЗАДЕРЖКИ ПКС-КОНТРОЛЛЕРА НА ПАРАЛЛЕЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ

Карташевский В.Г.¹, Галич С.В.², Семёнов Е.С.², Кирьянова Н.И.²

¹Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ

²Волгоградский государственный университет, Волгоград, РФ

E-mail: kartash@psati.ru

В данной статье авторами рассматриваются метрики и закономерности параллельных вычислений применительно к оценке масштабирования задержки контроллера программно-конфигурируемых сетей. Проанализирован закон Амдала для многоядерных центральных процессоров, поддерживающих распараллеливание на уровне потоков, и определены его области применимости. Проведены экспериментальные исследования масштабирования задержки контроллера программно-конфигурируемых сетей OpenDaylight на параллельной вычислительной системе, снабженной процессором Intel Xeon. В результате была выявлена тенденция к росту доли линейного кода при увеличении числа ядер центрального процессора. Данное явление ограничивает рост ускорения на параллельной вычислительной системе. Сделан вывод о том, что активация технологии распараллеливания потоков Hyper-Threading позволяет снизить задержку контроллера OpenDaylight и повысить эффективность использования процессора сервера. Результаты исследования могут быть полезны сетевым инженерам, администрирующим контроллеры программно-конфигурируемых сетей, а также программистам, разрабатывающим программное обеспечение для данного контроллера.

Ключевые слова: программно-конфигурируемые сети, ПКС-контроллер, OpenDaylight, задержка, многоядерный процессор, распараллеливание на уровне потоков, масштабирование, ускорение, закон Амдала

Введение

Внедрение в эксплуатацию сетей, реализующих идеологию разделения уровня управления трафиком от уровня передачи данных, получившей название программно-конфигурируемые сети (ПКС), порождает вопросы соблюдения оператором установленных норм функционирования IP-сетей [1-4]. Существует ряд работ [5-9], показывающих наличие зависимости параметров производительности сети в целом от производительности единого центра управления сетью, называемого в рамках данной концепции ПКС-контроллером и представляющего собой программно-аппаратную платформу на базе сервера общего назначения, использующего центральные процессоры (ЦП) архитектуры x86. В архитектуре современных ЦП положена идея параллельных вычислений: процессоры снабжаются несколькими вычислительными ядрами, а каждое ядро может обеспечить параллелизм на уровнях инструкций, обработки данных и потоков [10]. В случае

необходимости увеличения производительности аппаратной платформы эксплуатируемого ПКС-контроллера системный администратор может активировать технологию распараллеливания на уровне потоков, которая в случае процессоров Intel именуется Hyper-Threading, либо осуществить замену ЦП на модель с большим числом вычислительных ядер. Однако перед принятием таким мер необходимо убедиться в том, что эффект будет положительный для конкретного приложения [11-12]. В рамках данной статьи авторами рассматривается задача по выявлению закономерностей, позволяющих оценить масштабирование задержки ПКС-контроллера на параллельной вычислительной системе.

Метрики и закономерности параллельных вычислений

Для оценки работы программы на параллельной вычислительной системе часто используют такие метрики параллелизма, как ускорение и

эффективность. Рассмотрим каждую из этих метрик.

Ускорение – это отношение времени, затрачиваемого на выполнение задачи однопроцессорной системой, ко времени выполнения этой же задачи n -процессорной системой. Обозначим через n количество процессоров в вычислительной системе; V – объем вычислительных операций; W – скорость вычислений одним процессором; $T(n)$ – время выполнения задачи на n процессорах. Тогда время выполнения вычислений на однопроцессорной системе может быть определено по формуле $T(1) = V/W$.

На практике ни одну из задач невозможно распараллелить полностью, поскольку существуют фрагменты задачи, выполнить которые можно только последовательно, например ввод/вывод данных. Следовательно, любая задача, выполняемая на многопроцессорной системе, имеет последовательную и параллельную части. Пусть lp – время выполнения последовательной части решаемой задачи, тогда pp – время выполнения параллельно части. Очевидно, что $pp + lp = 1$. Тогда время выполнения вычислений на n -процессорной системе можно определить:

$$T(n) = \frac{lpV}{W} + \frac{ppV}{nW}. \quad (1)$$

Тогда, согласно данному выше определению, ускорение определяется как

$$S(n) = \frac{T(1)}{T(n)}. \quad (2)$$

Подставляя $T(n)$ и $T(1)$ в (2), получим

$$S(n) = 1 / (lp + \frac{pp}{n}). \quad (3)$$

Выражение (3) является математической записью закона [13-14], полученного Дж. Амдалом, известным архитектором компьютерных систем. Отметим, что данный закон имеет ряд ограничений. Считается, что объем решаемой задачи фиксирован ($V = const$), а выигрыш от дополнительных процессоров заключается в снижении времени вычисления. В законе Амдала не учитывается работа с памятью и издержки коммуникации между процессорами, а также он применим для равночастотных ЦП одинаковой архитектуры [13-14].

В 2008 г. М. Хилл и М. Марти в работе [15] предложили математическую запись закона Амдала для многоядерных процессоров и собственную методику классификации конфигурации

ЦП. Они ввели понятие «основной ядерный эквивалент» (Base Core Equivalent – далее BCE), в состав которого входят все ресурсы процессора, которые можно отнести собственно к ядрам (общие для ядер ресурсы в состав BCE не включаются). В зависимости от способа группировки BCE на чипе можно выделить процессоры симметричной (гомогенной), асимметричной (гетерогенной) и динамической конфигурации. Наиболее популярные серверные процессоры Intel Xeon можно отнести к процессорам с симметричной конфигурацией, при которой каждое ядро состоит из одного BCE [16].

Рассмотрим математическую запись закона Амдала для многоядерных симметричных процессоров. Пусть k – общее число BCE, из которых могут быть скомпонованы ядра; r – число BCE, составляющих ядро ЦП; $m = k/r$ – число ядер ЦП; $perf(r) = \sqrt{r}$ – условная производительность одного BCE; lp, pp – доли последовательной и параллельной части программы. Закон Амдала в записи Хилла-Марти имеет следующий вид:

$$S(m, r, lp) = \frac{1}{\frac{lp}{perf(r)} + \frac{pp \times r}{perf(r) \times k}}. \quad (4)$$

Стоит сказать, что ограничения закона Амдала распространяются и на его запись, предложенную Хиллом и Марти: в математическом выражении не учитывается работа с памятью и издержки коммуникаций между ядрами, а также разница тактовых частот ядер ЦП.

Для определения линейной доли кода программы (4) преобразуется к виду

$$lp = (\frac{m \times perf(r)}{S} - 1) / (m - 1). \quad (5)$$

Расчет линейной доли кода имеет смысл осуществлять при $m > 1$, так как в случае использования одноядерного процессора значение ускорения $S = 1$. При использовании многоядерных процессоров симметричной конфигурации с ядрами, состоящими из одного BCE каждое, коэффициент $perf(r) = 1$.

Для учета в записи закона Амдала (4) технологии Hyper-Threading добавим коэффициент, описывающий ускорение от распараллеливания на уровне потоков, рассчитываемый как отношение времени выполнения задачи на ЦП без Hyper-Threading ко времени выполнения этой же задачи на ЦП с Hyper-Threading при равном числе ядер:

$$S_{HT} = T_m / T_{mHT} . \quad (6)$$

Подставляя (6) в (4), получим запись закона Амдала для многоядерных процессоров с Hyper-Threading:

$$S(m, r, lp)_{HT} = S_{HT} \times \frac{perf(r)}{lp + \frac{pp}{m}} . \quad (7)$$

Исходя из такой записи закона Амдала будем в дальнейшем понимать под линейной долей кода lp ту часть программы, которую можно выполнить параллельно на физических ядрах ЦП, не учитывая распараллеливание потоков. Тогда технология Hyper-Threading рассматривается как способ оптимизации выполнения кода на одном ядре ЦП.

Помимо ускорения, для оценки качества параллелизма весьма полезной метрикой является эффективность использования ядер процессора, определяемая как отношение ускорения к числу ядер:

$$E(m) = S / m . \quad (8)$$

В идеальном случае ускорение зависит только от числа ядер $S(m) = m$, то есть зависимость имеет линейный вид. Эффективность же должна стремиться к единице $E(m) \rightarrow 1$, однако в случае использования Hyper-Threading может и превысить это значение. Однако данные показатели качества параллельных вычислений в некоторой степени противоречат друг другу, поскольку с увеличением числа ядер ЦП можно получить прирост ускорения при, как правило, снижении эффективности.

Методика эксперимента и обоснование применимости закона Амдала

Одной из ключевых метрик производительности ПКС-контроллера, описанной в проекте Рекомендации Инженерного совета Internet (IETF) [17], является задержка (latency). Задержка есть время, необходимое контроллеру на обработку одного асинхронного сообщения, отправляемого для установки нового потока на коммутаторе. Единицами измерения являются миллисекунды. Чем меньше задержка, тем выше производительность контроллера [17]. При экспериментальном исследовании задержки для повышения точности получаемых результатов рекомендуется повторять каждый тест 10 раз, после чего высчитывать среднее из полученных значений [17].

На данный момент существует весьма ограниченное число утилит для тестирования произво-

дительности ПКС-контроллеров, находящихся в свободном доступе [18]. Для экспериментального исследования масштабирования задержки ПКС-контроллера была выбрана утилита Sbench, поскольку её возможностей достаточно для требуемой задачи. Принцип действия Sbench заключается в эмуляции OpenFlow-коммутаторов, генерирующих асинхронные OpenFlow-сообщения packet_in. Данный тип сообщений отправляется коммутатором в качестве запроса на установку нового правила в таблицу потоков. Sbench позволяет задавать длительность одного теста, количество тестов за один запуск, количество конечных узлов, подключенных к коммутатору, и количество коммутаторов в сети. Для измерения задержки контроллера предназначен режим latency, при котором каждый коммутатор отправляет контроллеру сообщение packet_in и дожидается ответа flow_mod на отправленный запрос, прежде чем отправить следующий. Таким образом, режим latency позволяет определить время обработки одного сообщения, или задержку.

Исходя из условий применимости закона Амдала, можно предположить, что данный закон может использоваться для теоретической оценки максимально достижимого ускорения задержки ПКС-контроллера, поскольку при использовании утилиты Sbench в режиме latency объем задачи зафиксирован: $V = const$ и включает в себя последовательность действий контроллера по обработке единичного запроса типа packet_in от OpenFlow-коммутатора.

Сценарий тестирования и описание тестового стенда

С использованием утилиты Sbench был разработан следующий сценарий тестирования задержки ПКС-контроллера. Sbench запускался в режиме latency, при этом в ходе теста количество эмулируемых конечных узлов и коммутаторов оставалось постоянным и равнялось 10^5 и 16 шт. соответственно.

Таблица 1. Аппаратная конфигурация тестового стенда

Компонент	ЦП	Объем ОЗУ, Гб	Операционная система
ПКС-контроллер	Intel Xeon CPU E3-1241 v3 3,5 ГГц (4 ядра, 8 потоков)	16	Ubuntu Server 14.04
Сервер-нагрузчик	Intel Core i5 4590 3,3 ГГц (4 ядра)	32	Windows 7 Professional x64

Длительность одного теста составляла 10 с, осуществлялось 10 запусков утилиты и выполнялось 10 тестов за каждый запуск. Переменным параметром являлось число активных ядер ЦП. Аппаратная конфигурация тестового стенда приведена в таблице 1.

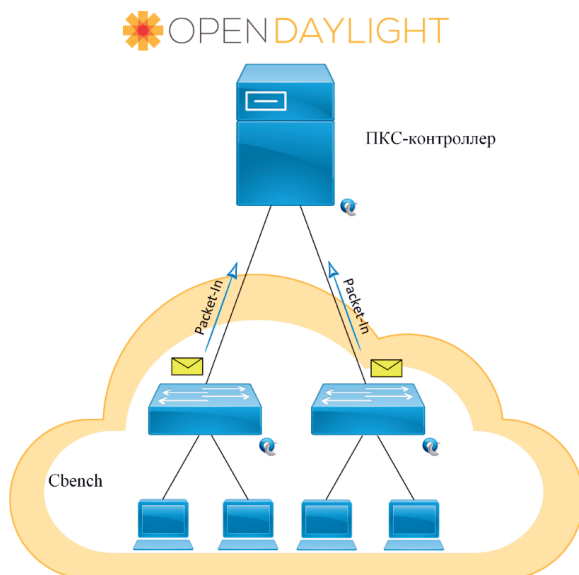


Рис. 1. Логическая схема экспериментального стенда

В роли ПКС-контроллера выступал OpenDaylight версии Beryllium-SR1. Контроллером использовалась версия Java 8.0 (платформа Java SE 1.8.0_101-b13). Нагрузочная утилита Cbench запускалась в виртуальной машине Oracle Virtual Box 5.0.20 в среде операционной системы Ubuntu 14.04. Виртуальной машине были выделены 4 ядра процессора и 24 Гб оперативной памяти сервера-нагрузчика. Оба сервера соединены каналом с пропускной способностью 1 Гбит/с. Для соблюдения условий выполнения закона Амдала технологию динамического повышения тактовой частоты ЦП Intel TurboBoost следует отключить в BIOS. Схема тестового стенда приведена на рис. 1.

Анализ результатов эксперимента

По результатам проведенного эксперимента были получены средние численные значения задержки обработки одного packet_in-запроса от каждого из 16 OpenFlow-коммутаторов. На основании значений задержки были рассчитаны ускорение, доля линейного кода OpenDaylight и эффективность. Результаты приведены в таблице 2 и на рис. 2-4.

Как видно из рис. 2, увеличение числа ядер ЦП приводит к снижению задержки, кроме того, активация технологии Hyper-threading позволяет дополнительно снизить значение задержки ПКС-контроллера. Характер зависимости ускорения от числа ядер,

в соответствии с рис. 3, далек от идеальной линейной функции, что обусловлено наличием весьма существенной доли нераспараллеливаемого кода.

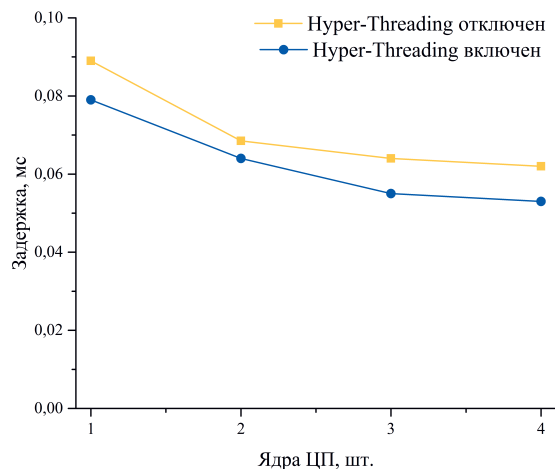


Рис. 2. Масштабирование задержки OpenDaylight с ростом числа ядер ЦП

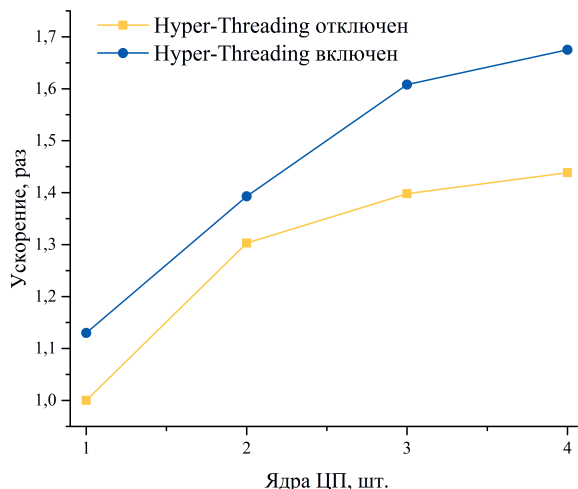


Рис. 3. Ускорение OpenDaylight с ростом числа ядер ЦП

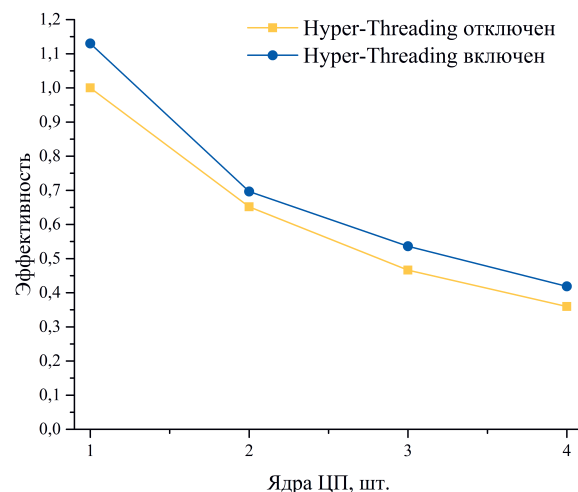


Рис. 4. Эффективность использования процессора контроллером OpenDaylight

Кроме того, как видно из таблицы 2, линейная доля кода OpenDaylight не является величиной постоянной, и при масштабировании числа ядер центрального процессора наблюдается ее рост. Объяснить данное явление можно возникновением накладных расходов [19], порождаемых как самой операционной системой, так и процессами

взаимодействия между ядрами, синхронизацией параллельных потоков [20].

Если же говорить об эффективности, то она, как видно из рис. 4, снижается с ростом числа ядер, однако активация технологии Hyper-Threading позволяет повысить эффективность использования ЦП.

Таблица 2. Результаты эксперимента

Количество ядер ЦП, шт.	1		2		3		4	
	–	+	–	+	–	+	–	+
Использование Hyper-Threading								
Задержка, мс	0,089	0,079	0,0685	0,064	0,064	0,055	0,062	0,053
Ускорение, раз	1	1,130	1,303	1,393	1,398	1,608	1,4385	1,675
Доля линейного кода	1		0,535		0,573		0,5935	
Эффективность	1	1,1302	0,6515	0,6965	0,4661	0,5361	0,3596	0,4187

Выводы

Проведенное на примере OpenDaylight исследование позволило показать применимость закона Амдала для описания масштабирования на многоядерной и многопоточной аппаратной платформе такой метрики производительности ПКС-контроллера, как задержка. Выявлено, что рост ускорения ограничивается долей линейного кода, которая к тому же имеет тенденцию к росту. Также интерес представляет изменение характера зависимости доли линейного кода от числа ядер при увеличении количества управляемых ПКС-контроллером OpenFlow-коммутаторов.

Таким образом, для прогнозирования масштабирования задержки при увеличении числа ядер центрального процессора необходимо ввести гипотезу о характере функциональной зависимости между линейной долей кода и количеством ядер ЦП, то есть решить задачу аппроксимации с помощью аналитического выражения. Это потребует изменения конфигурации тестового стенда, а именно смены ЦП на модель с большим числом ядер, что и будет являться направлением дальнейших исследований.

Кроме того, исследование показало, что активация технологии распараллеливания потоков Hyper-Threading способна снизить задержку контроллера и повысить эффективность

использования имеющихся аппаратных мощностей. Результаты исследования могут быть полезны сетевым инженерам и администраторам программно-конфигурируемых сетей, а также разработчикам программного обеспечения для контроллера OpenDaylight.

Литература

1. ITU-T Recommendation Y.1540. IP Packet Transfer and Availability Performance Parameters // ITU, 2016. – 57 p.
2. ITU-T Recommendation Y.1541. Network Performance Objectives for IP-Based Services // ITU, 2011. – 66 p.
3. ITU-T Recommendation Y.1291. An Architectural Framework for Support of Quality of Service in Packet Networks // ITU, 2004. – 30 p.
4. Об утверждении требований к организационно-техническому обеспечению устойчивого функционирования сети связи общего пользования // Приказ Минкомсвязи России № 113 от 27.09.2007.
5. Владыко А.Г., Матвиенко Н.А., Новиков М.И. и др. В. Тестирование SDN-контроллеров на базе модельной сети // Информационные технологии и телекоммуникации. Т. 4. № 1, 2016. – С. 17-28.
6. Владыко А.Г., Киричек Р.В., Великоречин М.А. и др. Комплексная методика тестирования фрагмента программно-кон-

- фигурируемой сети // Информационные технологии и телекоммуникации. № 2(10), 2015. – С. 20-29.
7. Семенов Е.С., Галич С.В., Тюхтяев Д.А. Анализ и классификация задержек, возникающих при работе протокола ARP в программно-конфигурируемых сетях // Вестник ГУ морского и речного флота им. С.О. Макарова. №5 (33), 2015. – С. 217-228.
 8. Галич С.В., Деогенов М.С., Карташевский В.Г. и др. Исследование производительности ПКС-контроллера OpenDaylight на сетях разных масштабов // Известия ЮФУ. Технические науки. №9, 2016. – С. 121-133.
 9. Azizi M., Benaini R., Mamoun M.B. Delay Measument in OpenFlow-enabled MPLS-TP Network // Modern Applied Science. Vol. 9, №3, 2015. – P. 90-101.
 10. Горшенин А.К., Замковец С.В., Захаров В.Н. Параллелизм в микропроцессорах // Системы и средства информатики. Т.24, №1, 2014. – С. 46-60.
 11. Guerin X., Tan W., Liu Y. e.a. Evaluation of Multicore Scalability Bottlenecks in Enterprise Java Workloads // Proceedings of the 2012 IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, August 07-09, 2012. – P. 308-317.
 12. Chen K.-Y., Chag J. M., Hou T.-W. Multithreading in Java: Performance and Scalability on MultiCore Systems // IEEE Transactions on Computers, December 02, 2010. – P. 1521-1534.
 13. Amdahl G.M. Validity of the single processor approach to achieving large scale computing capabilities // AFIPS'67 (Spring) Proceedings of the April 18-20, 1967, Spring Joint Computer Conference. New York: ACM, 1967. – P. 483-485.
 14. Амдал Дж. Компьютерная архитектура и закон Амдала // Открытые системы. №2. 2014. – С. 46-50.
 15. Hill M.D., Marty M.R. Amdahl's Law in the Multicore Era // Computer. Vol. 41 Issue 7, 2008. – P. 33-38.
 16. Martin C. Multicore processors: challenges, opportunities, emerging trends // Proc. Embedded World Conference 2014. Nuremberg, Germany. – P. 1-9.
 17. Vengainathan B., Basil A., Tassinari M. e.a. Internet-Draft Benchmarking Methodology for SDN Controller Performance / URL: <https://tools.ietf.org/html/draft-ietf-bmwg-sdn-controller-benchmark-meth-01> (д.о. 16.03.2016).
 18. Колечкин А.О., Владыко А.Г. Программное обеспечение для тестирования контроллеров программно-конфигурируемых сетей // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2016). М.: ноябрь, 2016.
 19. Багдасаров Г.А., Дьяченко С.В., Ольховская О.Г. Измерение производительности и масштабируемости программного комплекса MARPLE3D // Препринты ИПМ им. М.В.Келдыша. №37, 2012. – 22 с. / URL: <http://library.keldysh.ru/preprint.asp?id=2012-37> (д.о. 16.03.2016).
 20. Akhter S., Roberts J. Multi-Core Programming. Increasing Performance through Software Multi-threading. Intel Press, 2006. – 336 p.

Получено 11.04.2017

Карташевский Вячеслав Григорьевич, д.т.н., профессор, заведующий Кафедрой мультисервисных сетей и информационной безопасности Поволжского государственного университета телекоммуникаций и информатики. Тел. (8-846) 333-13-13. E-mail: kartash@psati.ru.

Галич Сергей Владимирович, аспирант Кафедры телекоммуникационных систем (ТКС) Волгоградского государственного университета (ВолГУ). Тел. 8-904-436-24-17. E-mail: sergeygali4@gmail.com

Семёнов Евгений Сергеевич, к.т.н., доцент, заведующий Кафедрой ТКС ВолГУ. Тел. 8-927-252-22-11. E-mail: esmenov@mail.ru

Кирьянова Нина Ивановна, к.э.н., доцент Кафедры ТКС ВолГУ. Тел. (8-844) 246-03-69. E-mail: volsu.tks@mail.ru

SCALABILITY ESTIMATION FOR SDN CONTROLLER LATENCY WHEN USING PARALLEL COMPUTING SYSTEM

Kartashevskii V.G.¹, Galich S.V.², Semenov E.S.², Kirianova N.I.²

*¹Povolzhskiy State University of Telecommunications and Informatics,
Samara, Russian Federation*

²Volgograd State University, Volgograd, Russian Federation

E-mail: kartash@psati.ru

This article studies software-defined networks controller latency scaling on parallel computing system. Metrics of parallel computing, such as speedup and efficiency, were determined. Amdahl's law for multicore processors with thread-level parallelism has been investigated. The range of applicability for Amdahl's law has been established. The test bed and the methodology of experiments have been discussed in details. OpenDaylight controller has been run on server with Intel Xeon central processor and its latency has been measured by Cbench utility. Herewith, the sequential fraction of OpenDaylight code as well as speedup and efficiency has been calculated. It is concluded that the sequential fraction of code is increasing during experiment with a growth of number of processor core and a more thorough study of the phenomenon should be performed. The findings also indicate that Hyper-Threading technology reduces software-defined network controller latency and improves efficiency of central processor usage.

Keywords: software-defined networks, SDN-controller, OpenDaylight, latency, multicore processors, thread-level parallelism, scalability, speedup, Amdahl's law

DOI: 10.18469/ikt.2017.15.2.09

Kartashevskii Viacheslav Grigorevich, Povolzhskiy State University of Telecommunications and Informatics, 23 L. Tolstoy str., Samara, 443010, Russian Federation; the Head of Department of Multiservice Networks and Information Security; Doctor of Technical Science, Professor. Tel.: +7846 3331313. E-mail: kartash@psati.ru

Galich Sergey Vladimirovich, Volgograd State University, 100 University Ave., Volgograd 400062, Russian Federation; PhD student of the Department of Telecommunication Systems. Tel.: +79044362417. E-mail: sergeygalich4@gmail.com.

Semenov Evgeniy Sergeevich, Volgograd State University, 100 University Ave., Volgograd 400062, Russian Federation; the Head of Department of Telecommunication Systems; PhD in Technical Science, Associate Professor. Tel.: +79272522211. E-mail: esemenov@mail.ru.

Kirianova Nina Ivanovna, Volgograd State University, 100 University Ave., Volgograd 400062, Russian Federation; Associate Professor of the Department of Telecommunication Systems; PhD in Economic Science. Tel.: +78442460369. E-mail: volsu.tks@mail.ru.

References

1. ITU-T Rec. Y.1540. IP Packet Transfer and Availability Performance Parameters. Geneva, Switzerland, ITU, 2016. 57 p.
2. ITU-T Rec. Y.1541. Network Performance Objectives for IP-Based Services. Geneva, Switzerland, 2011. 66 p.
3. ITU-T Rec. Y.1541. An Architectural Framework for Support of Quality of Service in Packet Networks. Geneva, Switzerland, 2004. 30 p.
4. Ministry of Telecom and Mass Communications of the Russian Federation (2007), Prikaz №113: Ob utverzhdenii trebovaniy k organizacionno-tekhnicheskomu obespecheniyu ustojchivogo funkcionirovaniya seti svyazi obshchego polzovaniya [Decree №113. On approval of the Requirements for organizational and technical support of the sustainable operation of the public telecommunication network]. Ministry of Telecom and Mass Communications, Moscow, Russia.
5. Vlyadyko A.G., Matvienko N.A., Novikov M.I., Kirichek R.V. Testirovanie SDN-kontrollerov nabaze model'noy seti [SDN - controllers benchmarking based on model network]. *Informacionnyye tehnologii i telekommunikacii*, 2016, vol. 4, no. 1, pp. 18-28.
6. Vlyadyko A.G., Kirichek R.V., Velikorechin M.A., Dumin D.I. Kompleksnaya metodika testirovaniya fragmenta programmno-konfiguriruemyy seti [Benchmarking methodology of software-defined networks]. *Informacionnyye tehnologii i telekommunikacii*, 2015, no. 2 (10), pp. 20-29. (In Russian)

7. Semenov E.S., Galich S.V., Tyukhtyaev D.A. Analiz i klassifikaciya zaderzhek voznikayushchih pri rabote protokola ARP v programmno-konfiguriruemym setyah [Analysis and classification of delays of ARP protocol in software-defined networks]. *Vestnik gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova*, 2015, no. 5, pp.217-228. doi: 10.21821/2309-5180-2015-7-5-217-228.
8. Galich S.V., Deogenov M.S., Kartashevskii V.G., Pasiuk A.O., Semenov E.S. Issledovanie proizvoditelnosti pks-kontrollera OpenDaylight na setyah raznykh masshtabov [OpenDaylight SDN controller: a study of performance scaling on networks of different sizes]. *Izvestiya Juzhnogo Federal'nogo universiteta. Tehnicheskie nauki*, 2016, no. 9, pp.121-134. doi: 10.18522/2311-3103-2016-9-121133.
9. Azizi M., Benaini R., Mamoun M.B. Delay Measurement in OpenFlow-enabled MPLS-TP Net-work. *Modern Applied Science*, 2015, vol. 9, no. 3, pp. 90-101. doi:10.5539/mas.v9n3p90.
10. Gorshenin A.K., Zamkovets S.V., Zakharov V.N. Parallelizm v mikroprocessorah [Parallelism in microprocessors]. *Systems and Means of Informatics*, 2014, vol. 24, no. 1, pp. 46-60. doi: 10.14357/08696527140102.
11. Guerin X., Tan W., Liu Y., Seelam S., Dube P. Evaluation of Multicore Scalability Bottlenecks in Enterprise Java Workloads. *Proceedings of the 2012 IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, VA, USA, 2012, pp. 308-317. doi:10.1109/MASCOTS.2012.43.
12. Chen K.-Y., Chag J. M., Hou T.-W. Multithreading in Java: Performance and Scalability on MultiCore Systems. *IEEE Transactions on Computers*, pp. 1521-1534. doi: 10.1109/TC.2010.232.
13. Amdahl G.M. Validity of the single processor approach to achieving large scale computing capabilities. *AFIPS '67 (Spring) Proceedings of the spring joint computer conference*. New York: ACM, 1967. pp. 483-485. doi: 10.1145/1465482.1465560
14. Amdahl G. Kompyuternaya arhitektura i zakon Amdala [Architecture of computers and Amdahl's law]. *Otkrytye sistemy*, 2014, no. 2, pp. 46-50.
15. Hill M.D., Marty M.R. Amdahl's Law in the Multicore Era. *Computer*, 2008, vol. 41, issue 7, pp. 33-38. doi: 10.1109/MC.2008.209.
16. Martin C. Multicore processors: challenges, opportunities, emerging trends. *Proc. Embedded World Conference 2014*, Nuremberg, Germany, pp. 1–9.
17. Vengainathan B., Basil A., Tassinari M. *Internet-Draft Benchmarking Methodology for SDN Controller Performance*. Available at: <https://tools.ietf.org/html/draft-ietf-bmwg-sdn-controller-benchmark-meth-01> (accessed 16.03.2016).
18. Kolechkin A.O., Vladyko A.G. [Software for testing of controllers in software-defined net-works]. *Materialy 19 Mezhdunarodnoy Nauchnoy Konferencii «Raspredelemnnye kompyuternye i telekommunikacionnye seti: upravlenie, vychislenie, svyaz»* [Proc. 19th Int. Scient. Conf. «Distributed computer and communication networks: control, computation, communications»]. Moscow, 2016, pp. 256-264.
19. Bagdasarov G.A., Dyachenko S.V., Olhovskaya O.G. [Program package MARPLE3D performance and scalability measurements]. *Preprint No.37 Keldysh Institute of applied mathematics*, Moscow, 2012. (In Russian). Available at: <http://library.keldysh.ru/preprint.asp?id=2012-37> (accessed 16.03.2016)
20. Akhter S., Roberts J. *Multi-Core Programming. Increasing Performance through Software Multi-threading*. USA, Intel Press, 2006. 336 p.

Received 11.04.2017

ТЕХНОЛОГИИ РАДИОСВЯЗИ, РАДИОВЕЩАНИЯ И ТЕЛЕВИДЕНИЯ

УДК 621.3.061

ИССЛЕДОВАНИЕ ЛОКАЛИЗАЦИИ ПИКОВЫХ ЗНАЧЕНИЙ СИГНАЛА В ПЕЧАТНОЙ ПЛАТЕ СИСТЕМЫ АВТОНОМНОЙ НАВИГАЦИИ

Газизов Р.Р., Газизов Т.Т.

Томский государственный университет систем управления и радиоэлектроники, Томск, РФ

E-mail: timurtsk@gmail.com

Рассмотрены особенности распространения сверхкоротких импульсов (СКИ) и локализации пиковых значений напряжений вдоль многопроводной шины печатной платы (ПП) разными подходами: с помощью ручного подбора длительности СКИ и при помощи оптимизации. Проведено моделирование распространения СКИ в форме тра-