

## ОЦЕНКА ПРОИЗВОДИТЕЛЬНОСТИ ПРИЛОЖЕНИЙ ПАРАЛЛЕЛЬНОЙ ОБРАБОТКИ ИЗОБРАЖЕНИЙ

Волотовский С.Г.<sup>1</sup>, Казанский Н.Л.<sup>1,2</sup>, Попов С.Б.<sup>1,2</sup>, Серафимович П.Г.<sup>1</sup>

<sup>1</sup> Учреждение Российской академии наук Институт систем обработки изображений РАН,

<sup>2</sup> Самарский государственный аэрокосмический университет имени академика С.П. Королёва  
(национальный исследовательский университет)

### Аннотация

Предложен метод оценки производительности приложений, предназначенных для параллельной обработки изображений. Данный метод базируется на сочетании аналитического подхода и использования результатов вычислительных экспериментов. Аналитическая составляющая метода представлена в характерном для типовых задач параллельной обработки изображений виде и параметрически учитывает три вида вычислительных затрат: использование процессора, межпроцессорные пересылки данных, операции ввода/вывода. Результаты вычислительных экспериментов использованы для оценки неизвестных параметров аналитической модели, что позволяет эффективным образом строить новые или модифицировать существующие алгоритмы параллельной обработки изображений.

**Ключевые слова:** параллельные вычисления, обработка изображений, масштабируемость параллельных приложений.

### Введение

Создание эффективных параллельных приложений предполагает умение предсказывать зависимость их производительности от алгоритма, внутренних параметров, архитектуры вычислительной системы. Для этой цели используются как аналитические подходы, так и результаты вычислительных экспериментов. Использование аналитики позволяет наиболее быстро получать требуемые результаты. Однако для многих сложных параллельных приложений затруднительно создать адекватную аналитическую модель оценки производительности. С другой стороны, проведение вычислительных экспериментов иногда требует затрат недопустимо большого времени [1].

Часто основное внимание при оценке производительности параллельного приложения уделяется той его компоненте, которая связана непосредственно с вычислениями на центральном процессоре. В этом случае, однако, исключение из анализа других важных компонентов, например, обмена данными между процессорами или операций ввода/вывода, может исказить правильность полученных результатов. Доля влияния каждого из компонентов параллельного приложения на производительность системы может зависеть от специфики самого приложения, архитектуры вычислительного комплекса, его поколения [2].

Например, в современных суперкомпьютерах и кластерах всё более увеличивается разрыв между производительностью центрального процессора и устройств ввода/вывода. Производительность центрального процессора растёт значительно быстрее. Чтобы оптимизировать операции ввода/вывода, необходимо уметь точно оценивать их вклад в общую работу системы. Учитывая иерархичность архитектуры современных средств работы с памятью, это является сложной задачей. Архитектура памяти обычно состоит из регистров центрального процессора, кэшей разных

уровней, оперативной и дисковой памяти. Скорость обмена между оперативной и дисковой памятью минимальна в этой иерархии. Поэтому оптимизация операций этого звена, как правило, наиболее важна [3].

С ростом объёма регистрируемой фото- и видеoinформации возрастает важность эффективной обработки крупноформатных изображений [4-6]. Для хранения крупноформатных изображений обычно используются файловые хранилища с возможностью параллельного чтения/записи. Возникающие при обработке изображения промежуточные результаты можно сохранять в локальной файловой системе (ЛФС) вычислительного узла или использовать сетевую файловую систему (СФС), например NFS (Network File System) [7]. Каждый из этих подходов имеет свои преимущества и недостатки.

Использование СФС является единственным способом сохранения промежуточных данных при использовании бездисковых кластеров. Преимуществом СФС является простота использования, при которой достигается доступность всего файла, размещённого в СФС, всем вычислительным узлам. Недостатком СФС является низкая по сравнению с ЛФС скорость чтения/записи. При активном использовании СФС также необходимо, чтобы файловые операции не замедляли операции пересылки данных между процессорами. Поэтому для файловых операций в СФС желательно организовать отдельную коммуникационную сеть между вычислительными узлами.

Использование ЛФС обеспечивает высокую скорость чтения/записи. Недостатком ЛФС для разработчика параллельного приложения является необходимость обеспечивать вычислительным узлам доступ к требуемым фрагментам используемого файла. Часто это приводит к дублированию частей файла на вычислительных узлах. Таким образом, по сравнению с СФС возрастает объём необходимого дискового пространства. Также использование ЛФС за-

трудняет организацию корректного завершения программы, при котором не нарушалась бы целостность сохранённых данных. Эта проблема особенно актуальна при использовании менеджера ресурсов кластера, например, Torque [8].

### Аналитическая модель

В качестве примера использования покомпонентного подхода, предлагаемого для оценки производительности приложений параллельной обработки изображений, рассмотрим задачу параллельной обработки изображения скользящим окном [9, 10]. Отсчёты обработанного изображения являются результатом свёртки соседних значений каждого отсчёта со значениями маски. Таким образом, при размере маски  $m \times m$  для каждого отсчёта выполняется  $m^2$  операций умножения-сложения. Будем полагать, что  $m$  нечётно. Определим время работы процессора, требуемое для выполнения одной операции умножения-сложения, как  $t_{CPU}$ . Тогда один проход по изображению скользящим окном требует времени при последовательной обработке  $m^2 t_{CPU} n^2$ . Здесь полагается, что изображение, как и маска, является квадратным и имеет размер  $n^2$ .

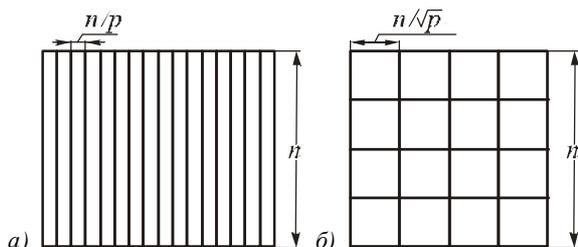


Рис. 1. Примеры распределения фрагментов изображения между процессорами: одномерная сетка (а) и двумерная сетка (б)

Параллельная обработка изображения предполагает распределение вычислений по заданному количеству процессоров  $p$ . На рис. 1 показаны два возможных способа такого распределения: разделение изображения на полосы (рис. 1а) и на квадратные блоки (рис. 1б). Оценим отношение затрат на пересылку данных между процессорами к вычислительным затратам для обоих этих способов. Обозначим этот параметр как ССР (communication-to-computation ratio) [11]. При разбиении изображения по одномерной сетке (рис. 1а) каждый процессор обрабатывает фрагмент изображения размером  $n^2/p$  отсчётов. Для обработки граничных отсчётов каждому процессору потребуются данные от соседних процессоров. Данная пересылка содержит  $n(m-1)$  отсчётов. Если не учитывать затраты на чтение/запись обрабатываемого изображения, то ССР равно  $2(m-1)p/n$ .

При разбиении изображения по двумерной сетке (рис. 1б) каждый процессор также обрабатывает фрагмент изображения размером  $n^2/p$  отсчётов. В данном случае пересылаются отсчёты, которые расположены по периметру каждого некраевого фрагмента изображения. Данная пересылка содержит

$4(m-1)n/\sqrt{p}$  отсчётов. Без учёта затрат на чтение/запись обрабатываемого изображения ССР в данном случае равно  $(4(m-1)\sqrt{p})/n$ .

Определим время работы параллельного алгоритма обработки изображения скользящим окном при разбиении изображения по одномерной сетке. Процесс обработки в общем случае состоит из 3 шагов:

- чтение/запись изображения,
- пересылка данных между процессорами,
- обработка фрагмента изображения скользящим окном.

Таким образом, время выполнения параллельного алгоритма обработки изображения скользящим окном записывается в следующем виде:

$$T_p = t_{CPU} \frac{km^2 n^2}{p} + k(m-1)(t_L + t_B n) + (t_{OC} + t_{RW} (n^2/p)), \quad (1)$$

где  $n \times n$  – размер изображения,  $m \times m$  – размер скользящего окна,  $k$  – количество проходов по изображению скользящим окном,  $p$  – количество процессоров,  $t_{CPU}$  – время выполнения базовой операции процессором,  $t_L$  – латентность межпроцессорных соединений,  $t_B$  – ширина пропускающей полосы межпроцессорных соединений,  $t_{OC}$  – время открытия/закрытия файла,  $t_{RW}$  – ширина полосы передачи данных при чтении/записи файла.

Функция накладных расходов, которая определяет время, потраченное всеми процессорами на «неполезную работу», в данном случае записывается в виде:

$$T_o = kp(m-1)(t_L + t_B n) + pt_{OC} + t_{RW} n^2. \quad (2)$$

Параметры, которые содержатся в аналитической модели (1), позволяют охватить широкий класс алгоритмов обработки изображений скользящим окном. Например, параметр  $m$  позволяет определить несколько видов операций:

- точечные операции ( $m = 1$ ),
- окрестные операции ( $1 < m < n$ ),
- глобальные операции ( $m = n$ ) [12].

### Результаты экспериментов

Для определения значений параметров в формуле (1), соответствующих конкретной вычислительной системе, было проведено моделирование на кластере, содержащем 28 четырёхядерных процессора ( $p = 28 \times 4 = 112$ ), с пиковой мощностью около 1 Тфлопс.

Для определения значения  $t_{CPU}$  фиксировались остальные (кроме  $n$ ) значения параметров в первом слагаемом формулы (1). При запуске тестовой программы обработки изображения скользящим окном замерялось время, затрачиваемое непосредственно на обработку, т.е. без учёта затрат на чтение/запись файла и на пересылку данных между процессорами. Размер отсчёта изображения равен одному байту.

На рис. 2а показаны затраты на обработку в виде зависимости времени от размера изображения. Точ-

ки на графике данной зависимости аппроксимируются линейной зависимостью. Исходя из этой аппроксимации, получаем  $t_{CPU} = 8,9 \text{e-}8$ .

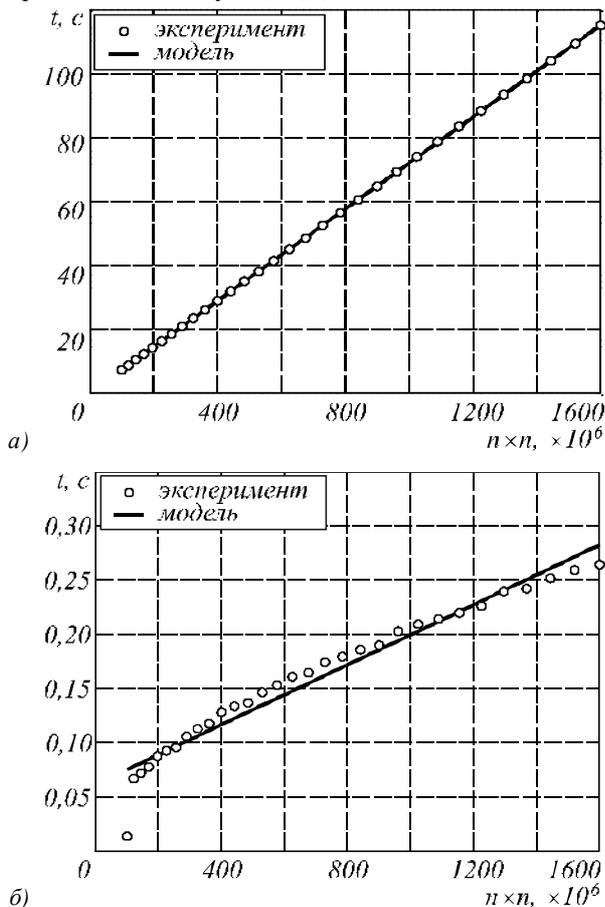


Рис. 2. Линейные аппроксимации зависимости времени выполнения фрагментов тестовой программы (секунд) от размера изображения (млн. отсчетов) для обработки изображения скользящим окном (а) и операций чтения/записи изображения (б)

Аналогично рассчитываем время, которое затрачивается на чтение изображения (рис. 2б). В исследуемой реализации алгоритма скользящего окна фрагменты обрабатываемого изображения размещаются локально на каждом вычислительном узле. Также, используя линейную аппроксимацию, получаем  $t_{RW} = 1,5 \text{e-}8$  и  $t_{OC} = 6,1 \text{e-}8$ .

На используемом кластере установлены коммуникационные сети двух спецификаций: Gigabit Ethernet и InfiniBand. На основании стандартных тестов для этих сетей были получены значения  $t_L = 46,85 \text{e-}6$ ,  $t_B = 9,5 \text{e-}6$  для Gigabit Ethernet и  $t_L = 1,25 \text{e-}6$ ,  $t_B = 8,9 \text{e-}7$  для InfiniBand. В дальнейшем анализе использованы параметры сети InfiniBand, если не оговорено другое.

Введённое выше понятие CCR удобно использовать для оценки вклада каждого из компонентов вычислительной системы в выполненную работу по параллельной обработке изображения. Определим понятие коммуникации в CCR как совокупные операции по чтению/записи обрабатываемого изобра-

жения и операции пересылки данных между процессорами. Тогда выражение для CCR запишется в виде

$$CCR = \frac{p(m-1)(t_L + t_B n)}{t_{CPU} m^2 n^2} + \frac{p t_{OC} + t_{RW} n^2}{k t_{CPU} m^2 n^2}. \quad (3)$$

На рис. 3а показана зависимость CCR от отношения  $t_{RW}/t_{CPU}$  для операций чтения изображения при различных размерах маски. Здесь фиксировалось значение  $t_{CPU} = 8,9 \text{e-}8$ . Можно интерпретировать уменьшение отношения  $t_{RW}/t_{CPU}$  при фиксированном  $t_{CPU}$  как использование более быстрых дисковых накопителей, а его увеличение как, например, использование сетевой файловой системы NFS вместо локального размещения фрагментов изображения на каждом вычислительном узле. С другой стороны, уменьшение отношения  $t_{RW}/t_{CPU}$  при фиксированном  $t_{RW}$  можно интерпретировать как использование более быстрых процессоров.

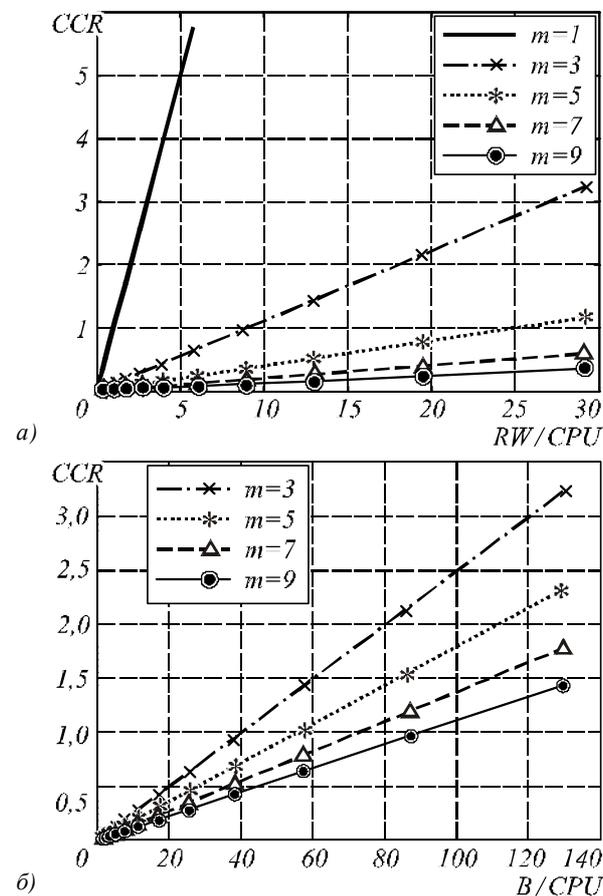


Рис. 3. Зависимость CCR для операций чтения файла от отношения  $t_{RW}/t_{CPU}$  (а), для операций пересылки от отношения  $t_B/t_{CPU}$  (б)

На рис. 3б показана зависимость CCR от отношения  $t_B/t_{CPU}$  для операций чтения изображения при различных размерах маски. Здесь также фиксировалось значение  $t_{CPU} = 8,9 \text{e-}8$ . Можно интерпретировать уменьшение отношения  $t_B/t_{CPU}$  при фиксированном  $t_{CPU}$  как использование более быстрых ком-

муникационных сетей. Так, переход на графике от значения отношения  $t_B/t_{CPU}$ , равного  $\sim 100$ , к значению, равному  $\sim 10$ , означает переход от сети Gigabit Ethernet к сети InfiniBand.

#### Анализ масштабируемости приложения

Для дальнейшего анализа определим эффективность задачи  $E$  как отношение полученного ускорения к количеству процессоров. Функция изоэффективности [1] определяет скорость, с которой должен расти размер задачи с ростом количества процессоров, чтобы поддерживать фиксированную эффективность. Рассматриваемая вычислительная задача считается хорошо масштабируемой, если размер изображения должен расти линейно как функция от количества процессоров.

В нашем случае функция изоэффективности записывается в виде

$$t_{CPU} m^2 n^2 = \frac{E}{1-E} T_o(n, p). \quad (4)$$

Чтобы получить зависимость размера изображения от количества процессоров, необходимо решить квадратичное по переменной  $n$  уравнение.

На рис. 4 показаны функции изоэффективности для нескольких значений эффективности. Отметим, что накладные расходы в виде затрат на пересылку данных и чтение/запись изображения не повлияли на линейный характер функции изоэффективности. На рис. 4а размер маски равен 3. На рис. 4б показана функция изоэффективности для различных значений размера маски. Значение эффективности на рис. 4б принято равным 70%.

Функция изоэффективности позволяет выполнять анализ масштабируемости параллельной задачи в зависимости от различных параметров. Например, из анализа рис. 4б следует, что с ростом размера маски при фиксированном размере задачи требуется большее количество процессоров, чтобы поддерживать заданное значение эффективности. Таким образом, с увеличением размера маски масштабируемость задачи улучшается.

#### Заключение

Проведённый покомпонентный анализ вычислительной системы позволил получить функции ускорения, накладных расходов, масштабируемости и исследовать их зависимость от различных параметров, например, от размера окна обработки, количества итераций, доли операций ввода/вывода в общем количестве вычислений.

Сочетание аналитического подхода с использованием результатов вычислительных экспериментов, с одной стороны, позволяет сократить время, затрачиваемое на вычислительные эксперименты при анализе эффективности сложных приложений параллельной обработки изображений. С другой стороны, аналитическая составляющая модели позволяет эффективным образом строить новые или модифицировать существующие алгоритмы параллельной обработки изображений.

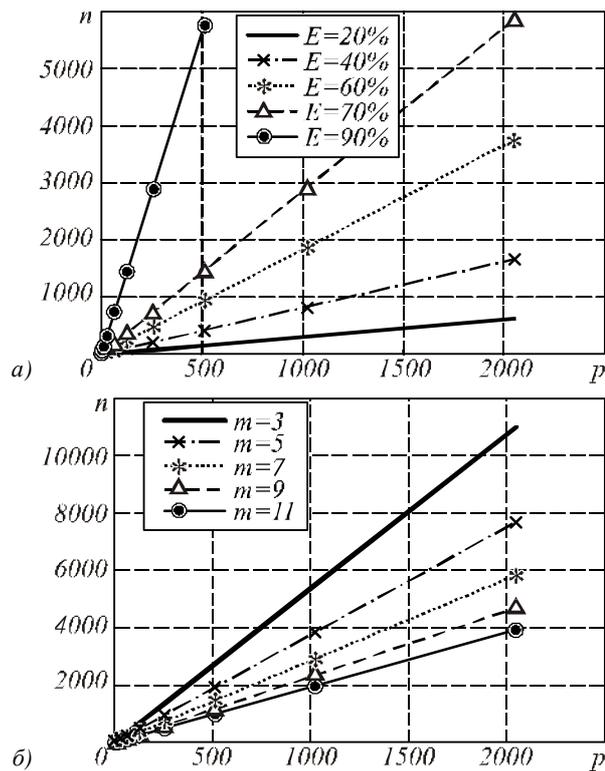


Рис. 4. Функция изоэффективности для  $m=3$  (а) различных значений размера маски  $m$  при  $E=70\%$  (б)

#### Благодарности

Работа выполнена при финансовой поддержке Программы фундаментальных исследований Президиума РАН «Проблемы создания национальной научной распределённой информационно-вычислительной среды на основе развития GRID технологий и современных телекоммуникационных сетей», грантов РФФИ № 10-07-00553, 09-07-00269 и гранта Президента РФ № НШ-7414.2010.9, а также Федеральной целевой программы «Научные и научно-педагогические кадры инновационной России» на 2009-2013 годы.

#### Литература

1. **Bagrodia R.** Performance Prediction of Large Parallel Applications Using Parallel Simulations / R. Bagrodia, E. Deelman, S. Doco and T. Phan // Proc. Seventh ACM SIGPLAN Symp. Principles and Practices of Parallel Programming (PPoPP '99). – 1999. – P. 151-162.
2. **Nunez, A.** New techniques for simulating high performance MPI applications on large storage networks / A. Nunez, J. Fernandez, J.D. Garcia, F. Garcia and J. Carretero // The Journal of Supercomputing. – 2010. – Vol. 51, No. 1. – P. 40-57.
3. **Krietemeyer, M.** The PRIOMark Parallel I/O-Benchmark / M. Krietemeyer, D. Versick, D. Tavangarian // Proc. of IASTED'05. – 2005. – P. 21-27.
4. **Попов, С.Б.** Концепция распределённого хранения и параллельной обработки крупноформатных изображений / С.Б. Попов // Компьютерная оптика. – 2007. – Т. 31, № 4. – С. 77-85. – ISSN 0134-2452.
5. **Попов, С.Б.** Моделирование информационной структуры параллельной обработки изображений / С.Б. Попов // Компьютерная оптика. – 2010. – Т. 34, № 2. – С. 231-242. – ISSN 0134-2452.

6. **Gashnikov, M.V.** Software System for Transmitting Large-Size Images via the Internet / M.V. Gashnikov, N.I. Glumov, S.B. Popov, V.V. Segreyev, and E.A. Farberov // Pattern Recognition and Image Analysis. – 2001. – V. 11, No. 2. – P. 430-432.
7. **Callaghan, B.** NFS Illustrated / B. Callaghan. – Boston: Addison-Wesley, 2000.
8. <http://www.clusterresources.com/products/torque-resource-manager.php>
9. **Soifer, V.A.** Computer Image Processing, Part I: Basic concepts and theory / V.A. Soifer, editor // VDM Verlag Dr. Muller, 2010.
10. **Soifer, V.A.** Computer Image Processing, Part II: Methods and algorithms / V.A. Soifer, editor // VDM Verlag Dr. Muller, 2010.
11. **da Silva, F.** Scalability analysis of embarrassingly parallel applications on large clusters / F. da Silva, H. Senger // Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), IEEE International Symposium. – 2010. – P. 1-8.
12. **Johasz, Z.** An Analytical Method for Predicting the Performance of Parallel Image Processing Operations / Z. Johasz // The Journal of Supercomputing. – 1998. – V. 12 N. 1-2. – P. 157-174.

### References

1. **Bagrodia R.** Performance Prediction of Large Parallel Applications Using Parallel Simulations / R. Bagrodia, E. Deelman, S. Docy and T. Phan // Proc. Seventh ACM SIGPLAN Symp. Principles and Practices of Parallel Programming (PPoPP '99). – 1999. – P. 151-162.
2. **Nunez, A.** New techniques for simulating high performance MPI applications on large storage networks / A. Nunez, J. Fernandez, J.D. Garcia, F. Garcia and J. Carretero // The Journal of Supercomputing. – 2010. – Vol. 51, No. 1. – P. 40-57.
3. **Krietemeyer, M.** The PRIOMark Parallel I/O-Benchmark / M. Krietemeyer, D. Versick, D. Tavangarian // Proc. of IASTED'05. – 2005. – P. 21-27.
4. **Popov, S.B.** Concept of distributed storage and parallel processing of large images / S.B. Popov // Computer Optics. – 2007. – V. 31, No. 4. – P. 77-85. – (in Russian).
5. **Popov, S.B.** Modelling the task information structure in parallel image processing / S.B. Popov // Computer Optics. – 2010. – V. 34, No. 2. – P. 231-242. – (in Russian).
6. **Gashnikov, M.V.** Software System for Transmitting Large-Size Images via the Internet / M.V. Gashnikov, N.I. Glumov, S.B. Popov, V.V. Segreyev, and E.A. Farberov // Pattern Recognition and Image Analysis. – 2001. – V. 11, No. 2. – P. 430-432.
7. **Callaghan, B.** NFS Illustrated / B. Callaghan. – Boston: Addison-Wesley, 2000.
8. <http://www.clusterresources.com/products/torque-resource-manager.php>
9. **Soifer, V.A.** Computer Image Processing, Part I: Basic concepts and theory / V.A. Soifer, editor // VDM Verlag Dr. Muller, 2010.
10. **Soifer, V.A.** Computer Image Processing, Part II: Methods and algorithms / V.A. Soifer, editor // VDM Verlag Dr. Muller, 2010.
11. **da Silva, F.** Scalability analysis of embarrassingly parallel applications on large clusters / F. da Silva, H. Senger // Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), IEEE International Symposium. – 2010. – P. 1-8.
12. **Johasz, Z.** An Analytical Method for Predicting the Performance of Parallel Image Processing Operations / Z. Johasz // The Journal of Supercomputing. – 1998. – V. 12 N. 1-2. – P. 157-174.

## PERFORMANCE ANALYSIS OF IMAGE PARALLEL PROCESSING APPLICATIONS

*S.G. Volotovskiy<sup>1</sup>, N.L. Kazanskiy<sup>1,2</sup>, S.B. Popov<sup>1,2</sup>, P.G. Serafimovich<sup>1</sup>*

<sup>1</sup>*Image Processing Systems Institute RAS,*

<sup>2</sup>*S.P. Korolyov Samara State Aerospace University*

### Abstract

Analytical and simulation approaches used for performance estimation of image parallel processing applications. Analytical formulation consists of three components: CPU calculation, MPI communication and I/O operations. Simulations used for estimation of parameters of the analytical model. Isoefficiency analysis shows the character of scalability dependence of parallel image processing applications on mask size, iterations number and share of I/O operations.

**Key words:** parallel computing, image processing, scalability of parallel applications.

### Сведения об авторах



**Волотовский Сергей Геннадьевич**, 1959 года рождения, в 1984 году окончил Куйбышевский авиационный институт имени академика С.П. Королёва (КуАИ) по специальности «Прикладная математика», работает ведущим программистом в Учреждении Российской академии наук Институт систем обработки изображений РАН. Область научных интересов: разработка программного обеспечения расчёта и моделирования работы элементов дифракционной оптики.

**E-mail:** [sv@smr.ru](mailto:sv@smr.ru)

**Sergey Gennadjevich Volotovskiy** (b. 1959) graduated from Kuibyshev Aviation Institute named after academician S. P. Korolyov (KuAI) on a specialty “Applied mathematics”, works as the leading programmer in the Image Processing Systems Institute of the RAS. Research interests: software design, modeling of systems with diffractive optical elements.



**Казанский Николай Львович**, 1958 года рождения. В 1981 году с отличием окончил Куйбышевский авиационный институт (КуАИ, ныне – Самарский государственный аэрокосмический университет имени академика С.П. Королёва – СГАУ) по специальности «Прикладная математика». Доктор физико-математических наук (1996 год), профессор, работает заместителем директора по научной работе и заведующим лабораторией дифракционной оптики Института систем обработки изображений РАН (ИСОИ РАН), профессором кафедры технической кибернетики СГАУ. Руководитель научно-образовательного центра компьютерной оптики, созданного совместно ИСОИ РАН и СГАУ, заведующий базовой (СГАУ в ИСОИ РАН) кафедрой высокопроизводительных вычислений. Является членом международных научных обществ *SPIE* и *IAPR*. Казанский Н.Л. – специалист в области дифракционной оптики, математического моделирования, обработки изображений и нанопластики. В списке научных работ Н.Л. Казанского 240 статей, 7 монографий, 35 авторских свидетельств и патентов. Страница в интернете: <http://www.ipsi.smr.ru/staff/kazansky.htm>.

E-mail: [kazansky@smr.ru](mailto:kazansky@smr.ru)

**Nikolay Lvovich Kazanskiy** (b. 1958) graduated with honours (1981) from the S. P. Korolyov Kuibyshev Aviation Institute (presently, S. P. Korolyov Samara State Aerospace University (SSAU)), majoring in Applied Mathematics. He received his Candidate in Physics & Maths (1988) and Doctor in Physics & Maths (1996) degrees from Samara State Aerospace University. He is the vice-director for research and the head of Diffractive Optics laboratory at the Samara Image Processing Systems Institute of the Russian Academy of Sciences (IPSI RAS), holding a part-time position of professor at SSAU's Technical Cybernetics sub-department. He is the manager of the Research & Education Center of Computer Optics established jointly by SSAU and IPSI RAS, holding the chair of SSAU's base sub-department of High-Performance Computing at IPSI RAS. He is a SPIE and IAPR member. He is co-author of 240 scientific papers, 7 monographs, and 35 inventions and patents. His current research interests include diffractive optics, mathematical modeling, image processing, and nanophotonics. Homepage: <http://www.ipsi.smr.ru/staff/kazansky.htm>.



**Попов Сергей Борисович**, кандидат технических наук, доцент Самарского государственного аэрокосмического университета имени академика С.П. Королёва; старший научный сотрудник Учреждения Российской академии наук Институт систем обработки изображений РАН. Области научных интересов: моделирование, разработка и исследование программных средств распределённой и параллельной обработки крупноформатных изображений; разработка алгоритмов и программного обеспечения систем технического зрения; разработка алгоритмов повышения качества цветных слабоконтрастных изображений.

E-mail: [spop@smr.ru](mailto:spop@smr.ru)

**Sergey Borisovich Popov**, Ph.D.; Associate Professor of the Samara State Aerospace University; senior researcher at the Image Processing Systems Institute of RAS. His areas of research are parallel and distributed image processing, computer vision, color image processing.



**Серафимович Павел Григорьевич**, кандидат физико-математических наук; старший научный сотрудник Учреждения Российской академии наук Институт систем обработки изображений РАН. Области научных интересов: моделирование и проектирование нанооптических устройств, методы исследования фотонных кристаллов, разработка и исследование программных средств распределённой и параллельной обработки крупноформатных изображений.

E-mail: [serp@smr.ru](mailto:serp@smr.ru)

**Pavel Grigorievich Serafimovich**, Candidate in Physics and Mathematics; senior researcher at the Image Processing Systems Institute of RAS. His areas of research are nanooptics, simulation and design of photonic crystals, parallel and distributed image processing.

Поступила в редакцию 26 ноября 2010 г.