

РАЗНОСТНОЕ РЕШЕНИЕ ВОЛНОВОГО УРАВНЕНИЯ НА ГРАФИЧЕСКИХ ПРОЦЕССОРАХ С ПОВТОРНЫМ ИСПОЛЬЗОВАНИЕМ ПОПАРНЫХ СУММ ДИФФЕРЕНЦИАЛЬНОГО ШАБЛОНА

Д.Г. Воротникова^{1,2}, Д.Л. Головашкин^{1,2}

¹ Институт систем обработки изображений РАН – филиал ФНИЦ «Кристаллография и фотоника» РАН, Самара, Россия,

² Самарский национальный исследовательский университет имени академика С.П. Королёва, Самара, Россия

Аннотация

Работа посвящена развитию методики построения векторных алгоритмов для разностного решения на графических процессорах задачи дифракции. Применение подхода, основанного на повторном использовании попарных сумм дифференциального шаблона, при решении уравнения Даламбера позволило до трёх раз сократить длительность вычислений по сравнению с известными алгоритмами.

Ключевые слова: векторные алгоритмы, волновое уравнение, ускорение вычислений.

Цитирование: Воротникова, Д.Г. Разностное решение волнового уравнения на графических процессорах с повторным использованием попарных сумм дифференциального шаблона / Д.Г. Воротникова, Д.Л. Головашкин // Компьютерная оптика. – 2017. – Т. 41, № 1. – С. 134-138. – DOI: 10.18287/2412-6179-2017-41-1-134-138.

Введение

Разностное решение уравнений Максвелла (FDTD-метод [1]) на протяжении последних 15 лет является незаменимым инструментом математического моделирования в быстроразвивающихся областях естествознания, изучающих генерацию, распространение и приём электромагнитного излучения: нанофотоники [2], наноплазмоники [3], теории фотонных кристаллов [4] и др. Характеризуясь простой программной реализацией, общностью математической модели, естественностью подхода к её решению, FDTD-метод вместе с тем известен высокой требовательностью к системным ресурсам вычислительного устройства, препятствующей его широкому распространению за пределами сообщества пользователей суперкомпьютеров.

Решение указанной проблемы традиционно ищут в упрощении математической модели либо в переходе к вычислениям на распространённых графических процессорах (GPU). В качестве примера первого подхода уместно сослаться на разностное решение уравнения Даламбера [1, 5, 6], применение которого при известных условиях [7] (TM-случай) позволяет получать решение без потери точности по сравнению с классическим FDTD-методом, при снижении на треть требований к объёму памяти ЭВМ. Следование второму подходу [8, 9, 10] позволяет задействовать богатые вычислительные ресурсы современных видеокарт, содержащих тысячи исполнительных устройств (CUDA-ядер) и находящихся в распоряжении большинства исследователей. Признанным недостатком GPU является небольшой объём видеопамяти по сравнению с кластерными решениями [9], что делает актуальным сочетание обоих упомянутых подходов, а именно – разностное решение уравнения Даламбера на графических процессорах.

Признавая несомненный успех применения GPU при разностном решении уравнений Максвелла (так, в [8] иллюстрируется, как использование пакета B-CALM, ориентированного на видеопроцессор, позволяет в 30 раз сократить длительность расчётов по сравнению с пакетом MEEP, использующим обычный

кластер), необходимо отметить определённое замедление в развитии данного направления вычислительной математики. Переход к современной методике построения векторных алгоритмов, основанной на использовании «длинных» векторов, позволил лишь незначительно (в 1,4 раза [11]) ускорить вычисления по сравнению с B-CALM. Это обстоятельство связано с особенностями дифференциального шаблона классической схемы Yee [1], ограничивающими выбор варианта векторизации вычислений алгоритмом на основе операции *general x plus y* [12]), уступающим по ускорению алгоритму на основе повторного использования попарных сумм [11]. Вместе с тем, как будет показано далее, использование дифференциального шаблона для уравнения Даламбера [13] освобождает от упомянутого ограничения, что также свидетельствует об актуальности выбранного направления исследований.

1. Векторный алгоритм на основе повторного использования попарных сумм

Забываясь в первую очередь о демонстрации идеи построения векторного алгоритма и сравнении с FDTD-методом, авторы выбрали для их иллюстрации классическое двумерное однородное линейное нестационарное волновое уравнение [7].

$$\frac{\partial^2 U}{\partial t^2} = c^2 \left(\frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} \right), \quad (1)$$

где $U(t, y, 0)$ – напряженность электрического поля, c – скорость света в среде (в однородном пространстве – константа, в неоднородном – функция координат). На краях вычислительной области D ($0 \leq t \leq T$, $0 \leq y \leq L_y$, $0 \leq z \leq L_z$) устанавливаются граничные условия Дирихле $U(t, y, 0) = 0$ и $U(t, y, L_z) = 0$, $U(t, 0, z) = 0$ и $U(t, L_y, z) = 0$, соответствующие электрической стенке. В качестве начального условия принимается отсутствие поля в области при $t \leq 0$ ($U(0, y, z)$ и $\partial U(0, y, z) / \partial t = 0$). Далее, при $t > 0$ поле появляется из «жесткого» источника (*hard source*, [1]), расположенного в произвольном месте D .

Для конструирования сеточной области воспользуемся известной областью Yee для FDTD-метода [1],

исключив узлы, в которых определялись сеточные аналоги проекций магнитного поля H_y и H_z . Получим D_h как $\{(t_n, y_m, z_k): t_n = nh_t, n = 0, 1, \dots, K = T/h_t, y_m = mh_y, m = 0, \dots, M = L_y/h_y, z_k = kh_z, k = 0, \dots, N = L_z/h_z\}$.

Следуя за [13], заменим в (1) частные производные их разностными аналогами и запишем для квадратной области ($L_y = L_z, M = N, h_y = h_z = h, h_t = \tau$) следующую хорошо известную явную разностную схему:

$$\frac{U_{m,k}^{n+1} - 2U_{m,k}^n + U_{m,k}^{n-1}}{\tau^2} = -c^2 \left(\frac{U_{m-1,k}^n - 2U_{m,k}^n + U_{m+1,k}^n}{h^2} + \frac{U_{m,k-1}^n - 2U_{m,k}^n + U_{m,k+1}^n}{h^2} \right),$$

которую для производства расчётов удобно представить в итерационном виде:

$$U_{m,k}^{n+1} = \alpha (U_{m,k-1}^n + U_{m,k+1}^n - 4U_{m,k}^n + U_{m-1,k}^n + U_{m+1,k}^n + U_{m,k-1}^n) + 2U_{m,k}^n - U_{m,k}^{n-1}, \quad \alpha = \frac{c^2 \tau^2}{h^2}. \quad (2)$$

На той же сеточной области соответствующим образом дискретизируются краевые, начальные условия и «жёсткий» источник.

При конструировании длинновекторного алгоритма решения (2) выберем стратегию повторного использования попарных сумм, как принесшую успех в [11] при решении уравнения теплопроводности. Если для схемы Y_{ee} в силу особенностей строения сеточной области это невозможно, то после перехода к уравнению Даламбера, – вполне. Ведь дифференциальный шаблон на n -м слое по времени для (2) основан на паттерне «крест», аналогичном для разностного уравнения теплопроводности (рис. 1 из [11]), в то время как по сравнению со схемой Y_{ee} это совершенно разные шаблоны. Интересно при этом отметить, забегая вперёд, что решение (2) совпадает с результатами вычислений по схеме Y_{ee} в любом знаке и совершенно отличается от решения уравнения теплопроводности в силу разности математических моделей.

Разумеется, уравнение (2) можно переписать и в канонической форме А.А. Самарского, организовав векторизацию вычислений по ней, как это было сделано в [11] для уравнений теплопроводности и Максвелла, однако там же была показана предпочтительность декларируемого подхода с использованием попарных сумм на примере уравнения теплопроводности.

Пользуясь нотацией, введённой в [11,12], запишем векторный алгоритм решения (2) в следующем виде:

Алгоритм

% Заполнение вектора T попарными суммами:

1. $T(2:N(N-1)) = V(2:N(N-1)) + V(N+1:N \times N - 1)$;

% Сохранение значений U^n

2. $Z = V$;

% Подсчёт значений для следующего временного слоя U^{n+1} :

3. $V(N+2:N(N-1)-1) = \alpha(T(2:N(N-1)-1) + T(N+3:N(N-2)+N+1)) + \alpha_1 V((N+2:N(N-1)-1) + V(1(N+2:N(N-1)-1)))$; $\alpha_1 = 2-4\alpha$

% Восстановление граничных значений.

4. $V1 = Z$;

Вектор

$$V = (U_{0,0}^n, U_{0,1}^n, \dots, U_{0,N}^n, U_{1,0}^n, U_{1,1}^n, \dots, U_{1,N}^n, \dots, U_{N,0}^n, U_{N,1}^n, \dots, U_{N,N}^n)$$

формируется построчным «разворачиванием» значений сеточного аналога функции U на временном слое n , вектор $V1$ – на слое $n-1$. Вектор Z используется для сохранения значений сеточной функции при переходе на следующий слой, когда решение на слое $n+1$ (третья строка Алгоритма) пишется поверх решения на слое n .

Ускорение вычислений по данному алгоритму по сравнению с «коротковекторным» подходом из [14], предложенным для организации вычислений по FDTD-методу, должно обеспечиваться за счёт двух факторов.

1. Перехода к «длинным» векторам, что позволит более полно загрузить вычислительные мощности видеопроцессора. Действительно, если следовать [14] и векторизовать вычисления вдоль одного выбранного пространственного направления сеточной области, то придётся работать с векторами длины $\sim N$, в предложенном алгоритме производятся операции над векторами из $\sim N^2$ значений.

2. Повторного использования попарных сумм дифференциального шаблона, записываемых в вектор T на первом шаге алгоритма. Например, при вычислении $U_{1,1}^{n+1}$ по (2) используется $U_{1,2}^n + U_{2,1}^n$, эта же сумма необходима при нахождении значения $U_{2,2}^{n+1}$. При его отыскании нет нужды пересчитывать её второй раз, как это принято в [14].

Проблема восстановления граничных значений (строка 4) и её программное решение аналогичны для разностного решения уравнений теплопроводности и Даламбера. Её обсуждение для первого уравнения приводится в [13].

В отличие от алгоритма для решения уравнений Максвелла из [11], в предложенном снижено количество арифметических операций с векторами вдвое, что позволяет надеяться на двукратное ускорение вычислений. Вместо 4 операций saхru [12], используется одна, вместо 4 операций сложения – 3 такие операции. При этом производится дополнительное копирование векторов (строки 2 и 5 алгоритма) и восстановление граничных значений, что может снизить выигрыш во времени вычислений.

2. Вычислительные эксперименты

Предлагаемая работа является завершающей в цикле статей, начатом публикацией [11] и продолженным в [15]. В силу чего программно-аппаратная база для постановки вычислительных экспериментов (видеокарта NVIDIA GeForce GTX 660Ti, центральный процессор Intel Core i7-3770, операционная система Debian Wheezy с установленными драйверами CUDA Toolkit 4 и компилятором gcc), приёмы программной реализации (использование библиотеки CUBLAS для исполнения векторных операций) и па-

параметры вычислительных экспериментов остаются прежними.

На рис. 1 и в табл. 1 приведены сравнения длительности вычислений по различным реализациям разностного решения уравнений Максвелла и Даламбера. Каждое значение получено в результате усреднения по 250 замерам.

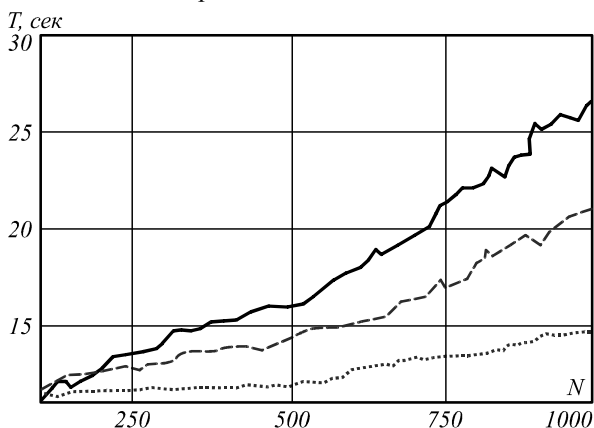


Рис. 1. Сравнение длительности вычислений по B-CALM [8] (сплошная линия) и длинновекторными алгоритмами: из [11] (пунктирная линия), основанным на векторизации операции гаура, и предложенным в настоящей работе (точечная линия), основанным на повторном использовании попарных сумм дифференциального шаблона

Табл. 1. Сравнение длительности вычислений по B-CALM (T_b), алгоритму из [11] (T_1) и предложенному алгоритму (T_2). Ускорения $S_1 = T_1/T_2$, $S_2 = T_b/T_2$

$N \times N$	T_1 , с	T_2 , с	S_1	T_b , с	S_2
50×50	2,53	2,47	1,10	2,41	0,96
100×100	6,01	2,71	2,21	4,36	1,6
150×150	6,92	2,80	2,47	6,87	2,45
200×200	7,87	3,03	2,59	7,92	2,61
250×250	8,73	3,36	2,58	11,53	3,43
300×300	9,51	3,67	2,6	13,31	3,62
350×350	10,27	3,95	2,61	14,37	3,64
400×400	11,20	4,31	2,59	15,68	3,63
450×450	12,12	4,66	2,60	16,30	3,49
500×500	13,83	5,90	2,34	17,45	2,96
650×650	18,03	8,09	2,22	22,64	2,8
800×800	20,42	10,3	2,10	24,14	2,34
1000×1000	23,86	13,05	1,82	28,50	2,18

Как и ожидалось, длительность вычислений по сравнению с алгоритмом из [11] снижена. Незначительно для небольшой сеточной области ($S_1 = 1,1$ при $N=50$), существенно для области средних размеров ($S_1 = 2,61$ при $N=350$) и в соответствии с прогнозом (двукратное уменьшение количества векторных операций) для крупных областей ($S_1 = 2,1$ и $1,82$ при $N=800$ и 1000).

Для средних и крупных областей более существенным оказался выигрыш авторского алгоритма по сравнению с пакетом B-CALM (табл. 1). К сожалению, программный код пакета, разработанного в Калифорнийском технологическом институте, явля-

ется достаточно запутанным, что затрудняет поиск причин достигнутого выигрыша.

Заключение

В настоящей работе предлагается векторный алгоритм разностного решения уравнения Даламбера, предназначенный для моделирования дифракции электромагнитного излучения в двумерном случае. Данным алгоритмом дополняется материал авторских публикаций [11, 15] и завершаются исследования по разработке векторных алгоритмов решения сеточных уравнений различных типов явных разностных схем (двух- и трёхслойных) для различных дифференциальных уравнений (переноса, теплопроводности и волнового), позволяющих ускорить вычисления на GPU по сравнению с популярными пакетами Open-Current (компании NVidia) и B-CALM.

Касаюсь рекомендаций по использованию алгоритмов из упомянутого цикла работ, авторы в первую очередь желают обратить внимание читателя на решение обратных задач дифракционной нанофотоники [2]. Действительно, использование для этой цели стохастических вычислительных процедур (например, генетического алгоритма из [2]) потребует многократного (десятки тысяч и более) решения прямой задачи дифракции. Тогда трёхкратное ускорение вычислений по сравнению с пакетом B-CALM позволит получить выигрыш по времени не в несколько секунд, как в табл. 1, а значительно более существенный. К тому же в ходе многочисленных вычислительных экспериментов была установлена (и подтверждена разработчиками B-CALM в личной переписке) невозможность использования указанного пакета для решения обратных задач в силу его невысокой надёжности.

Благодарности

Работа выполнена при частичной поддержке грантов РФФИ 14-07-00291-а и 16-47-630560-р_а.

Литература

1. **Taflove A.** Computational electrodynamics: The finite-difference time-domain method / A. Taflove, S. Hagness. – 3th ed. – Boston: Artech House Publishers, 2005. – 1006 p. – ISBN: 978-1-58053-832-9.
2. Дифракционная нанофотоника / А.В. Гаврилов, Д.Л. Головашкин, Л.Л. Досколович, П.Н. Дьяченко, А.А. Ковалёв, В.В. Котляр, А.Г. Налимов, Д.В. Нестеренко, В.С. Павельев, Р.В. Скиданов, В.А. Сойфер, С.Н. Хонина, Я.О. Шуюпова. – Под ред. В.А. Сойфера. – М.: Физматлит, 2011. – 680 с. – ISBN: 978-5-9221-1237-6.
3. **Климов, В.В.** Наноплазмоника / В.В. Климов. – М.: Физматлит, 2009. – 480 с. – ISBN: 978-5-922110-30-3.
4. **Lourtioz, J.-M.** Photonic crystals. Towards nanoscale photonic devices / J.-M. Lourtioz, H. Benisty, V. Berger, J.-M. Gerard, D. Maestre, A. Tchelekov, D. Pagnoux. - 2nd ed. - Berlin, Heidelberg: Springer-Verlag, 2008. - 514 p. – ISBN: 978-3-540-78346-6.
5. **Козлова, Е.С.** Моделирование распространения короткого двумерного импульса света / Е.С. Козлова, В.В. Котляр // Компьютерная оптика. – 2012. – Т. 36, № 2. – С. 158-164.
6. **Козлова, Е.С.** Моделирование предвестников Зоммерфельда и Бриллюэна в среде с частотной дисперсией на

- основе разностного решения волнового уравнения / Е.С. Козлова, В.В. Котляр // Компьютерная оптика. – 2013. – Т. 37, № 2. – С. 146-154.
7. **Неганов, В.А.** Линейная макроскопическая электродинамика. Том 1 / В.А. Неганов, С.Б. Раевский, Г.П. Яровой. - М: Радио и Связь, 2000. – 512 с. – ISBN 5-256-01505-2.
 8. **Wahl, P.** B-CALM: An open-source GPU-based 3D-FDTD with multi-pole dispersion for plasmonics / P. Wahl, D.-S. LyGagnon, Ch. Debaes, D.A.B. Miller, H. Thienpont // Optical and Quantum Electronics. – 2012. – Vol. 44, Issue 3. – P. 285-290. - DOI: 10.1007/s11082-012-9558-z.
 9. **Мальшева, С.А.** Реализация разностного решения уравнений Максвелла на графическом процессоре методом пирамид / С.А. Мальшева, Д.Л. Головашкин // Компьютерная оптика. – 2016. – Т. 40, № 2. – С. 179-187. - DOI: 10.18287/2412-6179-2016-40-2-179-187.
 10. **Zakirov, AV.** High performance FDTD code implementation for GPGPU supercomputers / A.V. Zakirov, V.D. Levenchenko, A.Yu. Perepelkina, Z. Yasunari // Keldysh Institute Preprints. – 2016. – No. 44. – 22 p. – DOI: 10.20948/prepr-2016-44-e.
 11. **Воротникова, Д.Г.** Алгоритмы с «длинными» векторами решения сеточных уравнений явных разностных схем / Д.Г. Воротникова, Д.Л. Головашкин // Компьютерная оптика. – 2015. – Т. 39, № 1. – С. 87-93.
 12. **Golub, G.H.** Matrix Computations / G.H. Golub, Ch.F. Van Loan. – 3rd ed. – Baltimore, London: Johns Hopkins University Press, 1996. – 694 p. – ISBN: 0-8018-5414-8.
 13. **Самарский, А.А.** Теория разностных схем / А.А. Самарский. - М.: Наука, 1977. - 656 с.
 14. **Anderson, E.** Performance of the CRAY multiprocessors / E. Anderson, J. Brooks, Ch. Grassel. - The Supplemental Performance Report, 1996. - 25 p.
 15. **Воротникова, Д.Г.** Моделирование вычислений на графических процессорах по разностным схемам / Д.Г. Воротникова, Д.Л. Головашкин, А.В. Кочуров // Компьютерная оптика. – 2015. – Т. 39, № 5. – С. 801-807. - DOI: 10.18287/0134-2452-2015-39-5-801-807.

Сведения об авторах

Воротникова Дарья Геннадьевна, научный сотрудник лаборатории дифракционной оптики Института систем обработки изображений РАН – филиала Федерального государственного учреждения Федеральный научно-исследовательский центр «Кристаллография и фотоника» Российской академии наук, ассистент кафедры нанотехнологий Самарского национального исследовательского университета имени академика С. П. Королева. Область научных интересов: FDTD-метод, векторные и матричные вычисления. E-mail: daryavorotnikova@gmail.com.

Головашкин Дмитрий Львович, доктор физико-математических наук, ведущий научный сотрудник лаборатории дифракционной оптики Института систем обработки изображений РАН – филиала Федерального государственного учреждения Федеральный научно-исследовательский центр «Кристаллография и фотоника» Российской академии наук, профессор кафедры прикладной математики Самарского национального исследовательского университета имени академика С. П. Королева. Область научных интересов: FDTD-метод, векторные и матричные вычисления, математическое моделирование оптических элементов и устройств нанофотоники. E-mail: dimitriy@smr.ru.

ГРНТИ: 27.41.41.

Поступила в редакцию 24 октября 2016 г. Окончательный вариант – 16 декабря 2016 г.

DIFFERENCE SOLUTIONS OF THE WAVE EQUATION ON GPU WITH REUSE OF PAIRWISE SUMS OF THE DIFFERENTIAL TEMPLATE

D.G. Vorotnikova^{1,2}, D.L. Golovashkin^{1,2}

¹ Image Processing Systems Institute of RAS – Branch of the FSRC “Crystallography and Photonics” RAS, Samara, Russia,

² Samara National Research University, Samara, Russia

Abstract

This work proposes a technique for constructing vector algorithms for solving the diffraction problem using a finite difference scheme on GPUs. The use of an approach based on the reuse of sums of the differential pattern when solving the D' Alembert equation allowed up to a three-fold reduction in the running time in comparison with the known algorithms.

Keywords: vector algorithms, wave equation, acceleration of computing.

Citation: Vorotnikova DG, Golovashkin DL. Difference solutions of the wave equation on GPU with reuse of pairwise sums of the differential template. Computer Optics 2017; 41(1): 134-138. DOI: 10.18287/2412-6179-2017-41-1-134-138.

Acknowledgements: The work was funded by the Russian Foundation of Basic Research grants Nos. 14-07-00291-a and 16-47-630560 p_a of.

References

- [1] Taflove A, Hagness S. Computational Electrodynamics: The Finite-Difference Time-Domain. Boston: Artech House Publishers; 2005. ISBN: 978-1-58053-832-9.
- [2] Soifer VA, ed. Diffraction nanophotonics [In Russian]. Moscow: “Fizmatlit” Publisher; 2011. ISBN: 978-5-9221-1237-6.
- [3] Klimov VV. Nanoplasmonics [In Russian]. Moscow: “Fizmatlit” Publisher; 2009. ISBN: 978-5-922110-30-3.

- [4] Lourtioz JM, Benisty H, Berger V, Gerard JM, Maystre D, Tchelnokov A, Pagnoux D. Photonic Crystals. Towards Nanoscale Photonic Devices. 2nd ed. Berlin, Heidelberg: Springer-Verlag; 2008. ISBN: 978-3-540-78346-6.
- [5] Kozlova ES, Kotlyar VV. Simulation of ultrafast 2D light pulse. *Computer Optics* 2012; 36(2): 158-164.
- [6] Kozlova ES, Kotlyar VV. Simulations of Sommerfeld and brillouin precursors in the medium with frequency dispersion using numerical method of solving wave equations. *Computer Optics* 2013; 37(2): 146-154.
- [7] Neganov VA, Raevskiy SB, Yarovoy GP. Linear macroscopic electrodynamics [In Russian]. Moscow: "Radio i Svyaz" Publisher; 2000. ISBN 5-256-01505-2.
- [8] Wahl P, LyGagnon DS, Debaes Ch, Miller DAB, Thienpont H. B-CALM: An open-source GPU-based 3D-FDTD with multi-pole dispersion for plasmonics. *Opt Quant Electron* 2012; 44(3): 285-290. DOI: 10.1007/s11082-012-9558-z.
- [9] Malysheva SA, Golovashkin DL. Implementation of the FDTD algorithm on GPU using a pyramid method. *Computer Optics* 2016; 40(2): 179-187. DOI: 10.18287/2412-6179-2016-40-2-179-187.
- [10] Zakirov AV, Levchenko VD, Perepelkina AYU, Zempo Yasunari High performance FDTD code implementation for GPGPU supercomputers. *Keldysh Institute Preprints* 2016; 44. DOI: 10.20948/prepr-2016-44-e.
- [11] Vorotnikova DG, Golovashkin DL. Long vectors algorithms for solving grid equations of explicit difference schemes. *Computer Optics* 2015; 39(1): 87-93.
- [12] Golub GH, Van Loan ChF. *Matrix Computations*. JHU Press; 1996. ISBN: 0-8018-5414-8.
- [13] Samarskiy AA. *The theory of difference schemes* [In Russian]. Moscow: Nauka; 1977.
- [14] Anderson E, Brooks J, Grassel Ch. Performance of the CRAY multiprocessors. *The Supplemental Performance Report*; 1996.
- [15] Vorotnikova DG, Golovashkin DL, Kochurov AV. Modeling of GPU computing using difference schemes. *Computer Optics* 2015; 39(5): 801-807. DOI: 10.18287/0134-2452-2015-39-5-801-807.

Author's information

Daria Gennadievna Vorotnikova, researcher Diffractive Optics lab at the Image Processing Systems Institute of RAS – Branch of the FSRC "Crystallography and Photonics" RAS. Works as assistant of Nanoengineering department at Samara University. Research interests are FDTD-method, vectors and matrix computation. E-mail: darya-vorotnikova@gmail.com.

Dimitriy Lvovich Golovashkin, Doctor of Physics and Mathematics, leader researcher Diffractive Optics laboratory at the Image Processing Systems Institute of RAS – Branch of the FSRC "Crystallography and Photonics" RAS. Works as professor of the Applied Mathematics department at Samara University. Research interests are FDTD-method, vectors and matrix computation, mathematical modeling of optical components and nanophotonic devices. E-mail: dimitriy@smr.ru.

Received October 24, 2016. The final version – December 16, 2016.