БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Новые методы электрокардиографии / Под ред. С.В.Грачева, Г.Г.Иванова, А.Л. Сыркина. Москва: Техносфера, 2007. – 552 с.
2. *Кулаичев А.П.* Компютерная электрофизиология и функциональная диагностика: Учеб. пособие. 4-е издание., перераб. и доп. – М.: ФОРУМ:ИНФРА-М, 2007. – 640 с.
3. Analog-Digital Conversion, Walt Kester, March 2004. Analog Devices , Inc.
4. Op amp applications, Walter G. Jung, 2002. Analog Devices , Inc.

**Синютин Евгений Сергеевич**
Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.
E-mail: kafmps@ttpark.ru.
347900, г. Таганрог, ул. Петровская, 81.
Тел.: 88634328052; 88634325825.

**Siniutin Evgeniy Sergeevich**
Taganrog Institute of Technological – Federal State-Owned Educational Establishment of Higher Vocational Education «Southern Federal University».
E-mail: kafmps@ttpark.ru.
81, Petrovskay street, Taganrog, 347900, Russia.
Phone: +78634328052; +78634325825.

УДК 519.6: 681.3

**Я.Е. Ромм, В.В. Забеглов**

# О ПАРАЛЛЕЛЬНОЙ ФОРМЕ ДИСКРЕТНОГО ПРЕОБРАЗОВАНИЯ ФУРЬЕ

*Изложена схема параллельного выполнения дискретного преобразования Фурье (ДПФ). Схема включает вычисление базиса ДПФ на основе кусочно-полиномиальной аппроксимации функций с помощью интерполяционного полинома Ньютона. При произвольно заданной априори границе погрешности временная сложность вычисления базиса составляет* $O(1)$. *Параллельное выполнение ДПФ, включая вычисление базиса, оценивается временной сложностью* $O(log_2 N)$.

*Дискретное преобразование Фурье.*

**Y.E. Romm, V.V. Zabeglov**

# ON THE PARALLEL FORM OF DISCRETE FOURIER TRANSFORM

*The article outlines the scheme of parallel performance of Discrete Fourier Transform (DFT). The scheme includes calculation of DFP base founded on piecewise polynomial approximation of functions by means of Newton interpolated polynomial. In case of a priori arbitrarily set bound of error, time complexity of the calculation constitutes* $O(1)$. *Parallel performance of DFT, including base calculation, is estimated as having time complexity* $O(log_2 N)$.

*Discrete Fourier Transform.*

**Постановка вопроса.** При реализации основных алгоритмов цифровой обработки сигналов (ЦОС) необходимо вычислять элементарные функции, составляющие базис ортогонального разложения. Для дискретного преобразования Фурье (ДПФ) базис составляют тригонометрические функции. Сложность и точность их вычисления в значительной мере влияют на быстродействие и точность ЦОС, поэтому к их вычислению предъявляются требования одновременно высокого быстродействия, вычислительной устойчивости и высокой точности. Ниже ставится задача синтезировать распараллеливаемый алгоритм вычисления базиса ДПФ с помощью кусочно-полиномиальной схемы аппроксимации функций. При этом схема основана на интерполяционном полиноме Ньютона, включает формулы редукции и распространяется на вычисление ДПФ в целом.

Параллельные алгоритмы рассматриваются на модели неветвящихся параллельных программ без учета обмена [3,6].

**Исходные соотношения ДПФ.** Фундаментальное уравнение для получения – $N$-точечного ДПФ записывается в виде

$$X(k)=\frac{1}{N}\sum_{n=0}^{N-1}x(n)e^{-j2\pi nk/N}=\frac{1}{N}\sum_{n=0}^{N-1}x(n)\left[cos(2\pi nk/N)-j\,sin(2\pi nk/N)\right]. \quad (1)$$

Здесь $X(k)$ – частотный выход ДПФ в $k$-ой точке спектра, где $k=0,1,\dots,N-1$; $N$ – число отсчётов при вычислении ДПФ; значение $x(n)$ представляет собой $n$-й отсчёт во временной области, $n=0,1,\dots,N-1$. Выходной спектр ДПФ $X(k)$ является результатом вычисления свёртки между выборкой, состоящей из входных отсчётов во временной области, и набором из $N$ пар гармонических базисных функций (косинус и синус).

**Таблично-алгоритмическая схема вычисления функций на основе интерполяционного полинома Ньютона.** Схема заимствована из [1]. Пусть рассматривается функция одной действительной переменной, для определенности вида

$$f(x)=sin(\pi x),\quad x\in[0,1]. \quad (2)$$

Вычисление функции (2) требуется для построения базиса преобразования Фурье. Выбирается система непересекающихся подынтервалов равной длины, объединение которых совпадает с $[0,1]$:

$$[0,1]=\bigcup_{i=0}^{P-2}[x_i,x_{i+1})\bigcup[x_{P-1},x_P],\ x_{i+1}-x_i=1/P,\ i=0,1,\dots,P-1. \quad (3)$$

При априори заданной границе $\varepsilon$ абсолютной погрешности аппроксимации функции (2) и для каждого отдельно взятого подынтервала из (3) строится интерполяционный полином Ньютона степени $n$, где $n$ выбирается минимальным для достижения заданной точности приближения одновременно на всех подынтервалах. При этом полином Ньютона преобразуется по дистрибутивности с приведением подобных так, что в результате преобразования на $i$-м подынтервале принимает канонический вид полинома от одной переменной:

$$P_n(x)=a_{0i}+a_{1i}x+a_{2i}x^2+\dots+a_{ni}x^n,\ x\in[x_i,x_{i+1}),\ i=0,1,\dots,P-1, \quad (4)$$

где согласно (3) для $i=P-1$ равенство сохраняется при $x=x_P$.

В (4) индекс коэффициента $i$ совпадает с номером подынтервала. Построение (4) выполняется для всех $P$ подынтервалов, чтобы на каждом из них не превышалась заданная граница погрешности:

$$|\,sin(\pi x)-P_n(x)\,|\le\varepsilon,\ x\in[x_i,x_{i+1}),\ i=0,1,\dots,P-1. \quad (5)$$

Если минимальное для всех подынтервалов $n$ найдено, то для функции (2) и для каждого подынтервала из (3) набор коэффициентов в (4) можно записать в память машины и сделать хранимым. В дальнейшем, когда потребуется вычислить функцию данного вида, вначале дешифрируется значение номера $i$ подынтервала, которое служит математическим адресом выборки коэффициентов (4). Если $x \in [x_i, x_{i+1})$, то $i = int\left(\frac{x}{H}\right)$, где int – целая часть числа, $H = x_{i+1} - x_i$. Порядок времени дешифрации оценивается как единичный $t = O(1)$. В дальнейшем время вычисления функции определяется степенью полинома (4). По схеме Горнера значение этого полинома вычисляется с временной сложностью $t(1) = n\left(t_y + t_c\right)$, где $t_c$, $t_y$ – время бинарного сложения и умножения соответственно. За счет уменьшения длины подынтервала степень $n$ в (4) можно сделать «сколь угодно» малой (но целой) при соответственном возрастании $P$ в (3). С целью осуществления описываемого построения вычисление коэффициентов начинается для значения $n=1$ при минимальном значении $P$. В случае невозможности достижения заданной точности (4) значение $P$ удваивается; это продолжается до нарушения допустимых границ значений $P$; при их нарушении снова делается переход к минимальному значению $P$, но уже при $n=2$, и т.д.

В рамках интерполяции по Ньютону схема минимизации степени полинома детализируется следующим образом. Если границы $i$-го подынтервала из (3) обозначить как $a_{i\,0}$, $b_{i\,0}$, шаг интерполяции – $w_i = \frac{b_{i0} - a_{i0}}{n}$, то равностоящие узлы интерполяции на текущем шаге можно записать в виде $x_{i\,j} = a_{i0} + j\,w_i$, $j = 0, 1, \ldots, n-1$. Полином Ньютона степени $n$ на $i$-м подынтервале для функции (3) и данных узлов интерполяции записывается в виде $\Psi_{ni}(x) = f(x_{i0}) + \sum_{j=1}^{n} \frac{\Delta^j y_{i0}}{j!\,w_i^j} \prod_{k=0}^{j-1} (x - x_{ik})$, где $\Delta^j y_{i0}$ – конечная разность $j$-го порядка в точке $x_{i0}$, или в обозначении $t = \frac{x - x_{i0}}{w_i}$,

$$\Psi_{ni}(t) = f(x_{i0}) + \sum_{j=1}^{n} \frac{\Delta^j y_{i0}}{j!} \prod_{k=0}^{j-1} (t - k)\,. \tag{6}$$

Процесс приведения (6) к виду, аналогичному (4), влечет значения коэффициентов $a_{0\,i} = f(x_{i0})$, $a_{l\,i} = \sum_{j=l}^{n} b_{i\,j} d_{l\,j}$, где $b_{i\,j} = \frac{\Delta^j y_{i0}}{j!}$, $d_{l\,j}$ – коэффициенты полиномов вида $P_{n\,j}(t) = d_{0\,j} + d_{1j}\,t + d_{2\,j}\,t^2 + \ldots + d_{n\,j}\,t^n$ с натуральными корнями, входящими в состав полинома Ньютона. Эти коэффициенты не зависят от вида аппроксимируемой функции, после вычисления их можно сделать хранимыми для всех использований интерполяции. Для получения коэффициентов используется аналог формул Виета из [4, 5].

Процесс построения полинома на каждом подынтервале из (3) начинается с $n=1$, $P=1$. Проверка на заданную соотношением (5) точность аппроксимации (где $P_{ni}(x)=\Psi_{ni}(t)$) выполняется на каждом подынтервале в проверочных точках, взятых для независимой переменной $t$. Если во всех проверочных точках каждого из подынтервалов соотношение (5) не нарушено, то аппроксимация на всех подынтервалах при данном значении $P$ осуществима полиномом $P_{ni}(x)=\Psi_{ni}(t)$ при выбранном $n$. Нарушение (5) в какой-либо точке при сохранении того же значения $P$ требует увеличить степень полинома на 1, после чего весь процесс построения $\Psi_n$ и проверки его на точность аппроксимации заново повторяется на всех подынтервалах. После достижения заданной погрешности $\varepsilon$ в (5) коэффициенты (4) заносятся в память компьютера по адресам $(i,\, l)$, где $i$ – номер подынтервала, $i=0,1,\dots,P-1$, $l$ – номер коэффициента полинома в (4), $l=0,1,\dots,n$, где $n$ – степень полинома.

**Вычисление базиса Фурье с учетом редукции аргумента к основному интервалу.** Выберем и зафиксируем количество точек $N$ в (1), для определённости положим, что $N$ – степень двойки. По схеме приведённой выше, вычисляется значения функции (2) в точках $x[j]=2j/N$, $j=0,1,\dots,\frac{N}{2}-1$. Каждой точке $x[j]$ с номером $j$ ставится в соответствие процессор с таким же номером. Процессоры работают синхронно и независимо друг от друга. Каждый процессор дешифрирует номер подынтервала $i=int\left(\frac{2j}{N\cdot H}\right)$, в который попала точка $x[j]$, затем считывает коэффициенты полинома (4), хранящиеся в памяти компьютера, по адресам $(i,\, l)$, и с помощью схемы Горнера (с включением правой границы промежутка изменения независимой переменной)

$$f\left(x[j]\right)=\left(\cdots\left(a_{ni}\,t+a_{(n-1)i}\right)t\;+a_{(n-2)i}\right)t+\ldots+a_{0\,i}$$

вычисляет значение функции в конкретной точке. Значения функции и её переменной записываются в память в виде $\left(x[j],\, f\left(x[j]\right)\right)$, где $j$ – номер ячейки. Временная сложность схемы составит

$$T(N/2)=O(1). \tag{7}$$

Результаты программной реализации схемы даны в табл. 1.

Таблица 1

**Степени интерполяционного полинома Ньютона, аппроксимирующего функцию $f(x)=sin(\pi x)$, $x\in[0,1]$ по таблично-алгоритмической схеме**

| $\varepsilon$ \ $k$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $10^{-4}$ | 6 | 5 | 4 | 3 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $10^{-5}$ | 8 | 6 | 4 | 4 | 3 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $10^{-6}$ | 8 | 7 | 5 | 4 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $10^{-7}$ | 10 | 7 | 6 | 5 | 4 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| $10^{-8}$ | 10 | 8 | 7 | 5 | 4 | 4 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |
| $10^{-9}$ | 12 | 9 | 7 | 6 | 5 | 4 | 4 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 |
| $10^{-10}$ | 12 | 10 | 8 | 6 | 5 | 5 | 4 | 4 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 1 |
| $10^{-11}$ | | 11 | 9 | 7 | 6 | 5 | 4 | 4 | 4 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 |
| $10^{-12}$ | | 11 | 9 | 8 | 6 | 6 | 5 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 |

*Окончание табл. 1*

| $\varepsilon$ \ $k$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $10^{-13}$ | | 12 | 10 | 8 | 7 | 6 | 5 | 5 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 |
| $10^{-14}$ | | | 10 | 9 | 7 | 6 | 6 | 5 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 | 2 |
| $10^{-15}$ | | | 11 | 9 | 8 | 7 | 6 | 5 | 5 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 2 | 2 |
| $10^{-16}$ | | | 12 | 10 | 8 | 7 | 6 | 6 | 5 | 5 | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 3 |
| $10^{-17}$ | | | 12 | 10 | 9 | 8 | 7 | 6 | 5 | 5 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 3 |
| $10^{-18}$ | | | | 11 | 10 | 8 | 7 | 6 | 6 | 5 | 5 | 4 | 4 | 4 | 3 | 3 | 3 | 3 |

Во входном столбце таблицы $\varepsilon$ обозначает априори задаваемую границу погрешности. Во входной строке $k$ задает показатель степени $P = 2^k$ для количества подынтервалов из (2). На пересечении строки, содержащей $\varepsilon$, и столбца, содержащего $k$, указывается минимальное значение степени $n$ интерполяционного полинома Ньютона, при которой функция в заголовке таблицы аппроксимируется полиномом данной степени с точностью до $\varepsilon$ на каждом из $P$ подынтервалов. Пустующая клетка означает, что в используемой версии языка программирования граница погрешности $\varepsilon$ оказалась недостижимой для данного числа подынтервалов ни при одном значении $n$.

Далее строится базис Фурье, точнее действительная, $cos(2\pi nk/N)$, и мнимая, $sin(2\pi nk/N)$, части, $n = 0, 1, \dots, N-1$, $k = 0, 1, \dots, N-1$, по значениям функции (2) в точках $x[j] = 2j/N$, $j = 0, 1, \dots, \frac{N}{2} - 1$. Для построения используется формула редукции синуса [4] $g(x) = sin(x)$:

$$\left.\begin{aligned} \sin(x) &= (-1)^K sign(x) \sin(\pi \cdot z), \\ N = \mathrm{int}\left(\frac{|x|}{\pi}\right)&, \quad z = \left\{\frac{|x|}{\pi}\right\}, \end{aligned}\right\}, \tag{8}$$

где $\{\alpha\}$ обозначает дробную, $int(\alpha)$ – целую часть числа $\alpha$. Схема строится следующим образом.

Каждой точке $x[n,k] = 2\pi nk/N$, $n = 0, 1, \dots, N-1$, $k = 0, 1, \dots, N-1$, ставится в соответствие процессор с индексами $(n, k)$. Процессоры, работая параллельно, вычислят значение $z[n,k] = \left\{\frac{|(2\pi nk)/N|}{\pi}\right\} = \left\{\frac{2nk}{N}\right\}$. Затем каждый процессор по значению $z[n,k]$ дешифрирует $x[j] = z[n,k]$ адрес $j$, $j = 0, 1, \dots, \frac{N}{2} - 1$, где в ячейке памяти с номером $j$ также хранится значение функции $f(x[j])$. Далее вычисляются значения $(-1)^K$, $K = int\left(\frac{|(2\pi nk)/N|}{\pi}\right) = int\left(\frac{2nk}{N}\right)$ и $sign(2\pi nk/N)$, эти значения перемножаются в соответствии с формулой (8). Таким образом, строится мнимая часть $sin(2\pi nk/N)$, $n = 0, 1, \dots, N-1$, $k = 0, 1, \dots, N-1$ базиса Фурье.

Затем (возможно, параллельно с тем, что ниже не учитывается) процессор с индексами $(n,k)$ в той же точке $2\pi nk/N$ вычисляет значение $cos(2\pi nk/N)$, $n=0,1,\dots,N-1$, $k=0,1,\dots,N-1$, т.е. действительную часть базиса Фурье. Действительная часть строится аналогично мнимой, отличие заключается лишь в том, что в точке $x[n,k]=2\pi nk/N$ значение $cos(2\pi nk/N)$ вычисляется с помощью формулы приведения $sin\left(\frac{\pi}{2}-\alpha\right)=cos(\alpha)$, а именно, множители, входящие в формулу (8), примут вид:

$$z=\left\{\frac{|\pi/2-(2\pi nk)/N|}{\pi}\right\}=\left\{\left|\frac{1}{2}-\frac{2nk}{N}\right|\right\},$$

$$K=int\left(\frac{|\pi/2-(2\pi nk)/N|}{\pi}\right)=int\left(\left|\frac{1}{2}-\frac{2nk}{N}\right|\right),\quad \beta=sign\left(\frac{\pi}{2}-\frac{2\pi nk}{N}\right).$$

Вычисленные значения $(\cos(2\pi nk/N),\sin(2\pi nk/N))$, $n=0,1,\dots,N-1$, $k=0,1,\dots,N-1$, записываются в ячейки памяти с адресами $(n,k,m)$, $m=0,1$. Значение $m=0$ соответствует значению $\cos(2\pi nk/N)$, $m=1$ – $\sin(2\pi nk/N)$. Временная сложность схемы составит

$$T\left(N^2\right)=O(1). \tag{9}$$

Таким образом, имеет место

**Теорема 1.** Базис ДПФ вида $cos(2\pi nk/N)-j\,sin(2\pi nk/N)$, $n=0,1,\dots,N-1$, $k=0,1,\dots,N-1$, можно вычислить с произвольно заданной априори границей погрешности $\varepsilon$ при помощи таблично-алгоритмической схемы вычисления функций на основе интерполяционного полинома Ньютона и формулы редукции синуса на $N^2$ процессорах с временной сложностью $O(1)$, иными словами, вычисление выполнимо с временной сложностью (9).

**Схема параллельного вычисления ДПФ.** Вычисляется действительная и мнимая часть ДПФ, а именно соотношение (1) рассматривается в следующем виде

$$Re(X(k))=\frac{1}{N}\sum_{n=0}^{N-1}x(n)\,cos(2\pi nk/N), \tag{10}$$

$$Im(X(k))=-\frac{1}{N}\sum_{n=0}^{N-1}x(n)\,sin(2\pi nk/N), \tag{11}$$

$n=0,1,\dots,N-1$, $k=0,1,\dots,N-1$. Вычислим действительную часть ДПФ (10). ДПФ вычисляется с помощью схемы сдваивания следующим образом. Каждому коэффициенту $Re(X(k))$, $k=0,1,\dots,N-1$ взаимно однозначно ставится в соответствие $N$ процессоров с номерами $(n,k,0)$, $n=0,1,\dots,N-1$, $k=0,1,\dots,N-1$. Процессор с номером $(n,k,0)$ считывает значение $cos(2\pi nk/N)$ из ячейки памяти с номером $(n,k,0)$. Процессоры, работая синхронно и независимо, параллельно найдут все произведения элементов входной последовательности $x(n)$ на соответствующие значения $cos(2\pi nk/N)$, затем по схеме сдваивания за $log_2 N$ шагов складываются парные произведения. Вычисленная сумма делится на $N$ . Временная сложность схемы составит

$$T\left(N^2\right) = O\left(log_2 N\right) . \qquad (12)$$

Мнимая часть ДПФ (11) вычисляется аналогично. Каждому коэффициенту $Im\left(X\left(k\right)\right)$, $k = 0, 1, \ldots, N-1$ взаимно однозначно ставится в соответствие $N$ процессоров с номерами $\left(n,k,1\right)$, $n = 0, 1, \ldots, N-1$, $k = 0, 1, \ldots, N-1$. Процессор с номером $\left(n,k,1\right)$ считывает значение $sin\left(2\pi nk/N\right)$ из ячейки памяти с номером $\left(n,k,1\right)$. Процессоры, работая синхронно и независимо, параллельно найдут все произведения элементов входной последовательности $x\left(n\right)$ на соответствующие значения $sin\left(2\pi nk/N\right)$, затем за $log_2 N$ шагов по схеме сдваивания складываются парные произведения. Вычисленная сумма делится на $-N$ .

Временная сложность схемы параллельного вычисления действительной и мнимой части ДПФ составит

$$T\left(2N^2\right) = O\left(log_2 N\right). \qquad (13)$$

Таким образом, имеет место

**Теорема 2.** В условиях теоремы 1 базис и собственно ДПФ (1) можно вычислить с произвольно заданной априори границей погрешности $\varepsilon$ на основе таблично-алгоритмической схемы и схемы сдваивания на $2N^2$ процессорах с временной сложностью $O\left(log_2 N\right)$, иными словами, полное вычисление реализуется с временной сложностью (13).

**Заключение.** Изложены схемы параллельного вычисления базиса ДПФ и вычисления ДПФ. Таблично-алгоритмическая схема построения базиса ДПФ имеет временную сложность $O\left(1\right)$ для произвольно заданной априори границы погрешности $\varepsilon$ , время вычисления базиса не зависит от числа отсчётов ДПФ. Число процессоров квадратично зависит от числа отсчётов как при построении базиса, так и при вычислении ДПФ. При этом схема вычисления ДПФ с той же границей погрешности имеет временную сложность $O\left(log_2 N\right)$.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК


1. *Аксайская Л. Н.* Разработка и исследование параллельных схем цифровой обработки сигналов на основе минимизации временной сложности вычисления функций / Автореферат диссертации на соискание ученой степени кандидата технических наук. – Таганрог: ТТИ ЮФУ. – 2008. – 18 с.
2. *Ахмед Н., Рао К.Р.* Ортогональные преобразования при обработке цифровых сигналов: Пер. с англ. Т.Э. Кренкеля / Под ред. И.Б. Фоменко. – М.: Связь, 1980. – 248 с.
3. *Миклошко Й.* Связь между алгоритмами, программами и структурой параллельных ЭВМ. – В кн.: Алгоритмы математического обеспечения и архитектура многопроцессорных вычислительных систем / Под ред. А.П. Ершова. – М.: Наука, 1982. – С. 6-36.
4. *Ромм Я.Е.* Бесконфликтные и устойчивые методы детерминированной параллельной обработки / Дис… д-ра техн. наук. – Таганрог: ТРТУ, 1998. – 546 с.; ВНТИ Центр. – № 05.990.001006.
5. *Ромм Я.Е.* Локализация и устойчивое вычисление нулей многочлена на основе сортировки. II // Кибернетика и системный анализ. – 2007. – № 2. – С.161-174.
6. *Солодовников В.И.* Верхние оценки сложности решения систем линейных уравнений // Теория сложности вычислений. 1: Записки научных семинаров ЛОМИ АН СССР. – Л., 1982. – Т. 118. – С. 159-187.

**Ромм Яков Евсеевич**
ГОУВПО «Таганрогский государственный педагогический институт».
E-mail: romm@list.ru.
347926, г. Таганрог, ул. Инициативная, 48.
Тел.: 88634601753, 88634601812; 88634601807.

**Забеглов Валерий Валерьевич**
E-mail: vzv.box@gmail.com.
347900, г. Таганрог, ул. Октябрьская, 84, кв. 61.
Телефон: 88634367933.

**Romm Yakov Evseevich**
Taganrog State Pedagogocal Institute.
E-mail: romm@list.ru.
48, Initsiativnaya street, Taganrog 347926, Russia.
Phone: +78634601753, +78634601812, +78634601807.

**Zabeglov Valeriy Valer'evich**
E-mail: vzv.box@gmail.com.
Ap. 61, 84, Oktyabrskaya street., Taganrog, 347900, Russia.
Tel: +78634367933.

УДК 681.3

**С.А. Зори, С.В. Ковальский**

## АВТОМАТИЗАЦИЯ СОЗДАНИЯ ТРЕХМЕРНЫХ МОДЕЛЕЙ РЕАЛЬНЫХ ЛАНДШАФТОВ НА ОСНОВЕ ФОТОГРАФИЙ

*Предложена модификация алгоритма реконструкции трехмерной модели ландшафтов по их фотографиям, исследованы характеристики предложенного алгоритма.*

*Ландшафт; реконструкция; модель; алгоритм; модификация.*

**S.A. Zori, S.V. Kowalski**

## AUTOMATION OF CREATION OF THREE-DIMENSIONAL MODELS OF REAL LANDSCAPES ON THE BASIS OF PHOTOS

*Modification of algorithm of landscapes- three-dimensional model reconstruction is offered, characteristics of the offered algorithm is given.*

*A landscape; reconstruction; model; algorithm; modification.*

**Введение.** Компьютерная графика, синтез и обработка изображений определяют сегодня уровень развития не только вычислительной техники в целом, но и достижения и эффективность решения множества важных прикладных задач из самых разнообразных областей. Современные навигационные системы, летные тренажеры, системы визуального моделирования и другие самые разнообразные системы виртуальной и дополненной реальности, которые были признаны [1] одной из технологий, которые будут оказывать наиболее сильное влияние на развитие IT-отрасли в ближайшие годы и плотно проникшие сегодня в повседневную деятельность человека, строятся с применением трехмерных виртуальных сред, основанных на реальных ландшафтах и характеристиках их субстанций.

**Создание трехмерных моделей реальных ландшафтов на основе фотографий.** При построении трехмерных виртуальных сред возникает задача автоматизации создания трехмерных моделей реальных ландшафтов на основе имеющегося фото- или видеоряда [2]. Задача реконструкции трехмерной сцены по имею-