

## РЕАЛИЗАЦИЯ БЫСТРОГО АЛГОРИТМА ПРЕОБРАЗОВАНИЯ КИРХГОФА НА ПРИМЕРЕ БЕССЕЛЕВЫХ ПУЧКОВ

С.А. Балалаев, С.Н. Хонина

Самарский государственный аэрокосмический университет,  
Институт систем обработки изображения РАН

### Аннотация

В данной статье рассматриваются высокоэффективные алгоритмы непараксиального вычисления комплексного поля лазерного пучка на различных расстояниях от источника. Проводится сравнительный анализ ранее используемых и вновь разработанных алгоритмов. Несмотря на общность применения разработанных алгоритмов, их эффективность достигается при соблюдении некоторых условий. Приведены характеристики точности и скорости алгоритмов, полученные при их применении в конкретных задачах.

### Введение

Решение задачи дифракции при непараксиальном распространении света в свободном пространстве остается актуальной проблемой. Основная сложность связана с большой вычислительной трудоемкостью реализации интегральных преобразований в оптике, вытекающей из быстроосциллирующего характера ядра этих преобразований. Известны алгоритмы, использующие быстрое преобразование Фурье (БПФ), однако при этом теряется информация о реальном масштабе получаемых распределений. К тому же, в некоторых случаях БПФ применить невозможно, например, при реализации интеграла Кирхгофа.

В данной работе рассмотрены свойства симметрии ядра интеграла Кирхгофа и разработан алгоритм вычисления, значительно сокращающий время расчета по сравнению с реализацией численного интегрирования «в лоб». Эффективность разработанного алгоритма исследована на примере бесселевых пучков.

### 1. Описание алгоритма

Метод Кирхгофа решения дифракционных задач состоит в использовании интегральной теоремы, согласно которой значение функции  $U(x, y, z)$  на выходе, являющейся решением скалярного уравнения Гельмгольца:

$$\frac{\partial^2 U(x, y, z)}{\partial x^2} + \frac{\partial^2 U(x, y, z)}{\partial y^2} + \frac{\partial^2 U(x, y, z)}{\partial z^2} + k^2 U(x, y, z) = 0 \quad (1)$$

можно выразить через значение функции  $U(\xi, \eta, 0)$  на входе [1]:

$$U(x, y, z) = -\frac{z}{2\pi} \iint_{\Sigma} U(\xi, \eta) \frac{e^{ikr}}{r^2} \left( ik - \frac{1}{r} \right) d\xi d\eta, \quad (2)$$

где

$$r = \sqrt{(x - \xi)^2 + (y - \eta)^2 + z^2}, \quad (3)$$

$k = 2\pi/\lambda$  - волновое число, а  $\lambda$  - длина волны светового пучка. Выражение (2) получается через разложение светового поля по сферическим волнам и

представляет собой несколько более точную форму интеграла Кирхгофа.

Для программной реализации (2) «в лоб» можно воспользоваться численным интегрированием по методу трапеций:

$$I = \iint_{\Sigma} F(\xi, \eta) d\xi d\eta = \frac{1}{4} S_d \sum_{n=2}^N \sum_{m=2}^N [F_{n-1, m-1} + F_{n-1, m} + F_{n, m-1} + F_{n, m}] \quad (4)$$

где  $S_d$  - элементарная площадь области между отчетами плоскости интегрирования.

Чтобы использовать (4) для реализации алгоритмов вычисления оператора (2), исходное  $U(\xi, \eta)$  и выходное  $U(x, y, z)$  поле разбивают по отчетам [2], реальные координаты которых вычисляются по сетке:

$$\begin{cases} \xi_n = D/N(n - N/2); & x_i = D'/N(i - N/2); \\ \eta_m = D/N(m - N/2), & y_j = D'/N(j - N/2), \end{cases} \quad (5)$$

где  $N \times N$  - размер входной и выходной картины по отчетам;  $D \times D$  - реальный размер входной и  $D' \times D'$  - выходной картин.

Воспользуемся формулой численного интегрирования (4) и разбиением входных и выходных масштабов сеткой (5) для выражений (2) и (3):

$$\begin{cases} F_{n, m} = U(\xi_n, \eta_m) \frac{e^{ikr}}{r^2} \left( ik - \frac{1}{r} \right), \\ z \text{ де } r = \sqrt{(x_i - \xi_n)^2 + (y_j - \eta_m)^2 + z^2}; \\ U(x_i, y_j, z) = -\frac{z}{2\pi} \frac{D^2}{4N^2} \times \\ \times \sum_{n=2}^N \sum_{m=2}^N [F_{n-1, m-1} + F_{n-1, m} + F_{n, m-1} + F_{n, m}]. \end{cases} \quad (6)$$

Система (6) является простейшей реализацией непараксиального оператора распространения света на расстояние  $z$  от источника. Основным недостатком является большая ресурсозатратность алгоритма. Вызвано это тем, что подынтегральная функция  $e^{ikr}$  из-за большого значения волнового числа  $k$  является быстроосциллирующей и приходится выбирать очень мелкий шаг интегрирования.

В работе [3] была произведена оценка шага интегрирования, переписывая ее для рассматриваемого случая, получаем:

$$h_r \leq \frac{\sqrt{R_D^2 + z^2}}{R_D} \frac{\lambda}{N_h}, \quad (7)$$

где  $R_D = D/2 + D'/2$ ,  $N_h$  - количество разбиений одного периода осцилляций  $T_{min}$  (для алгоритма (6) предполагалось  $N_h = 10$ , см. рис. 1).

Выделим ядро интегрального преобразования (2):

$$G_r = \frac{e^{ikr}}{r^2} \left( ik - \frac{1}{r} \right), \quad (8)$$

и будем использовать для него шаг разбиения не крупнее, чем (7).

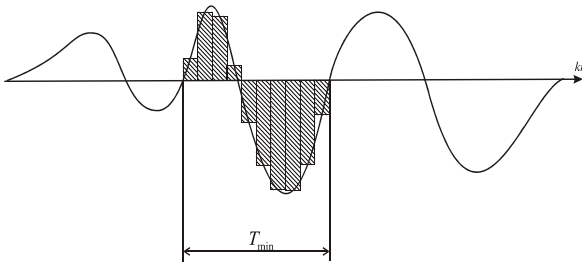


Рис. 1. Разбиение быстроосциллирующей функции  $G_r$  на области интегрирования, исходя из минимального периода биений  $T_{min}$

Необходимое число отсчетов не трудно вывести из (7):

$$M \geq \frac{1}{N} \frac{R_D}{\sqrt{R_D^2 + z^2}} \frac{N_h}{\lambda}. \quad (9)$$

Выражение (9) показывает, какое число разбиений функции  $G_r$  является достаточным в области одного отчета входной функции для ее правильного «прописывания». На рис. 2 представлены дополнительные разбиения функции  $G_r$  в области отсчета входной функции.

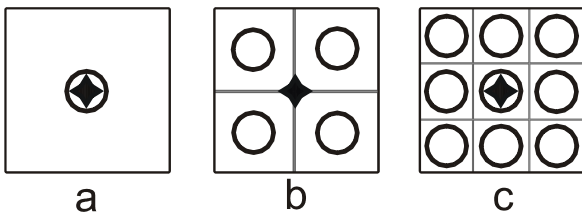


Рис. 2. Разбиение матрицы  $G$  на дополнительные отсчеты: (a) соответствует отсутствию разбиения ( $M=1$ ), (b) двойному разбиению ( $M=2$ ), (c) тройному ( $M=3$ )

Используя разбиение на дополнительные отсчеты (см. рис. 2) для (8) имеем:

$$r = \sqrt{(x_i + x'_k - \xi_n)^2 + (y_j + y'_l - \eta_m)^2 + z^2}, \quad (10)$$

где

$$\begin{cases} x'_k = \frac{D'}{NM} (k - (M+1)/2); \\ y'_l = \frac{D'}{NM} (l - (M+1)/2). \end{cases} \quad (11)$$

Теперь не трудно переписать (6) используя (8), (10), (11):

$$\begin{cases} G_{n,m} = \frac{1}{M^2} \sum_{k=1}^M \sum_{l=1}^M \frac{e^{ikr}}{r^2} \left( ik - \frac{1}{r} \right), \quad \text{где} \\ r = \sqrt{(x_i + x'_k - \xi_n)^2 + (y_j + y'_l - \eta_m)^2 + z^2}; \\ F_{n,m} = U(\xi_n, \eta_m) G_{n,m}; \\ U(x_i, y_j, z) = -\frac{z}{2\pi} \frac{D^2}{4N^2} \times \\ \times \sum_{n=2}^N \sum_{m=2}^N [F_{n-1,m-1} + F_{n-1,m} + F_{n,m-1} + F_{n,m}]. \end{cases} \quad (12)$$

При расчете выходного изображения легко заметить, что функция  $G_r$  периодически повторяется, поэтому можно сократить время выполнения алгоритма (12) при помощи таблицы  $G_{n,m}$ . Действительно, из рис. 3 видно, что матрица таблицы значений  $G_r$  имеет размерность  $2N-1$ , что значительно сокращает время вычисления. Матрица  $G$  полностью покрывает все значения матрицы  $F$ , необходимые для подсчета любого  $U(x_i, y_j, z)$ . На рис. 3 показано вычисление центрального значения  $U(0,0,z)$ , когда суммируются произведения соответствующих элементов заштрихованной области. Для вычисления произвольных значений  $U(x_i, y_j, z)$  соответственно матрица  $F$  перемещается по всей области матрицы  $G$ . Таким образом, вместо  $N^4$  получаем  $4N^2$  операций.

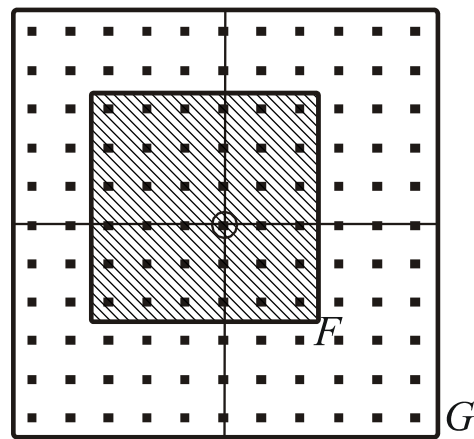


Рис. 3. Вычисление выходных значений при помощи перемножения соответствующих значений подынтегральной функции  $F$  на значения табулированной функции  $G$

Далее можно еще больше ускорить вычисление функции (8), воспользовавшись свойством радиальной симметрии  $G_r$ . На рис. 4 в заштрихованной области показаны элементы  $G_{n,m}$ , которые не повторяются в матрице  $G$ , это составляет 1/8 всей матрицы. Далее ис-

пользуется алгоритм радиальной развертки. В результате получаем оптимизацию вычисления в 8 раз.

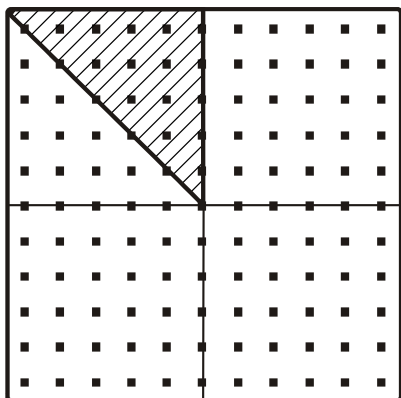


Рис. 4 Радиальная развертка матрицы  $G$

## 2. Исследование алгоритма

Рассмотрим теперь результаты работы алгоритма полученного на основе выражения (12), но реализованного различными способами – через пересчет всех значений (обычный) и с табуляцией. В качестве входной функции  $U(\xi, \eta)$  возьмем одномодовый беселевый пучок:

$$U(r, \varphi) = C_{m,\alpha} J_m(\alpha r) \exp(im\varphi), \quad (13)$$

сформированный ДОО размером  $5 \times 5$  мм, с параметрами моды  $m=1$ ,  $\alpha=7$ ,  $C_{m,\alpha}=1$  (амплитудно-фазовая картина входного распределения показана на рис. 5).

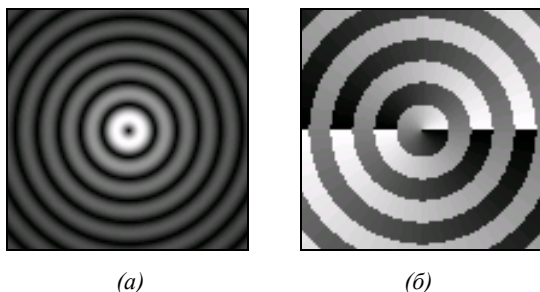


Рис. 5. Амплитуда (а) и фаза (б) моды Бесселя с параметрами:  $m=1$ ,  $\alpha=7$ ,  $C_{m,\alpha}=1$  во входной плоскости

Длину волны монохроматического лазерного пучка освещающего ДОО примем равной  $\lambda=633$  нм. В первую очередь нас будут интересовать результаты в непараксиальной области - на расстояниях, не превышающих  $z=70$  мм от плоскости ДОО. Выберем количество отчетов входных/выходных изображений  $N=100$ . Исходя из (9) примем нижний предел числа внутренних разбиений экспоненциальной функции  $M=23$ . Для заданного  $z$  получим две выходные функции  $U_0(x,y,z)$  и  $U(x,y,z)$ , представленные на рис.6. Одна соответствует алгоритму с пересчетом всех значений (верхняя строка), другая алгоритму с табуляцией (нижняя строка). Визуально они ни чем не отличаются, однако мы можем оценить их

различие с помощью среднеквадратического отклонения:

$$\delta = \sqrt{\frac{\iint_{\Sigma} (|U_0(x,y,z)| - |U(x,y,z)|)^2 dx dy}{\iint_{\Sigma} |U_0(x,y,z)|^2 dx dy}}, \quad (14)$$

где  $U_0(x,y,z)$  - комплексная амплитуда поля, вычисленная обычным алгоритмом численного интегрирования;  $U(x,y,z)$  - комплексная амплитуда поля, полученная оптимизированным методом. Для  $z=70$  мм по формуле (14) получаем  $\delta=10^{-11}$ , при этом на одной и той же машине оптимизированный без развертки алгоритм рассчитывается в 845 раз быстрее обычного, а с разверткой в 1440 раз быстрее. Причем погрешность между результатами, полученными двумя последними алгоритмами равна машинному нулю.

Далее произведем расчет погрешности вычислений по формуле:

$$\delta_M = \sqrt{\frac{\iint_{\Sigma} (|U_{2M}(x,y,z)| - |U_M(x,y,z)|)^2 dx dy}{\iint_{\Sigma} |U_{2M}(x,y,z)|^2 dx dy}}, \quad (15)$$

где  $U_M(x,y,z)$  - комплексная амплитуда поля, полученная с числом дополнительного разбиения  $M$ ,  $U_{2M}(x,y,z)$  - комплексная амплитуда поля с удвоенным числом разбиений. Для  $z=70$  мм значение (15) получилось равным  $\delta_M=0,05\%$ .

На рис. 6 и в таблице 1 представлены результаты численных экспериментов по распространению беселевого пучка, показанного на рис. 5. Они демонстрируют практическое совпадение результатов, полученных обоими алгоритмами на всей непараксиальной области. При этом с уменьшением расстояния от входной плоскости оптимизированный алгоритм демонстрирует все большую эффективность в экономии машинного времени, что наглядно показано в таблице 1.

На рис. 7 показаны аналогичные результаты, но для параксиальных областей, полученные теми же алгоритмами (в связи с отсутствием визуальных различий приведены изображения только для одного алгоритма). В этом случае выигрыш по времени менее существенный, однако даже для дальней зоны ( $z>2000$  мм) обеспечивается ускорение в 4 раза при различии, равном машинному нулю (см. таблицу 1).

Из таблицы 1 видно, что погрешность отлична от нуля там, где число дополнительных разбиений  $M>1$ , т.е. операция развертывания матрицы  $G$  вносит некоторое (несущественное) отличие в результаты. Таким образом, алгоритм с табуляцией позволяет вычислять интеграл Кирхгофа с той же точностью что и при использовании обычного алгоритма (если не использовать радиальную развертку) с ускорением не менее, чем в 4 раза. При использовании же дополнительных разбиений точность вычислений повышается.

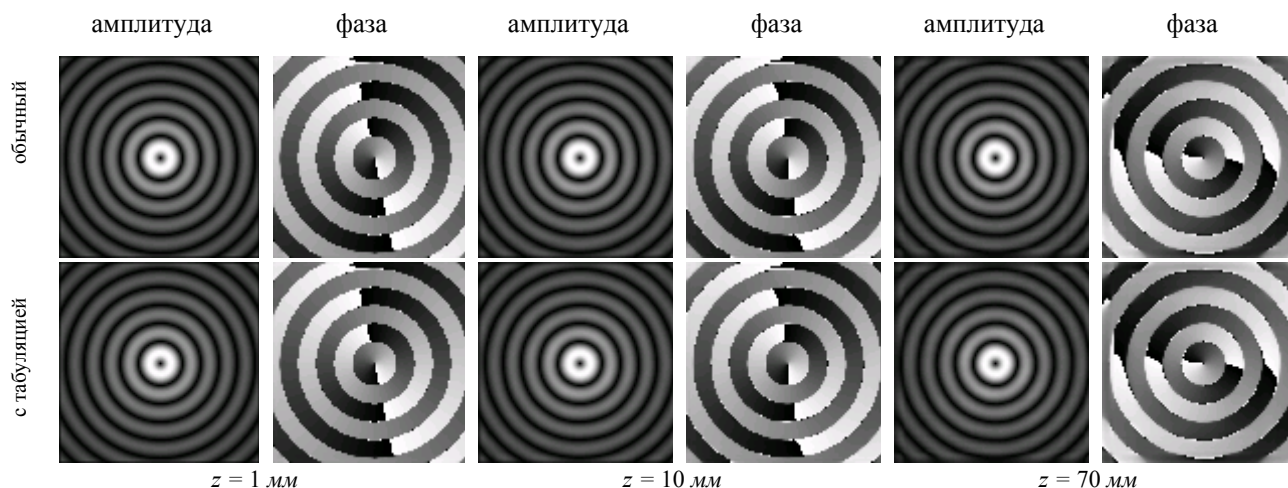


Рис. 6. Комплексное распределение моды Бесселя с параметрами:  $t=1$ ,  $\alpha=7$ ,  $C_{t,\alpha}=1$  на различных расстояниях от экрана при использовании обычного алгоритма (верхняя строка) и с таблицей (нижняя строка)

Таблица 1. Сравнение параксиальных и непараксиальных операторов

Расстояние от ДОО $z$ , мм	Число отсчетов входной и выходной функции, $N$	Число разбиений экспоненциальной функции, $M$	Время выполнения обычного алгоритма (час:мин:сек)	Время выполнения оптимизированного алгоритма (час:мин:сек)	СКО между амплитудами обычного и оптимизированного алгоритма $\delta$ , %	Ускорение в $K$ раз
1000	100	2	00:08:13	00:00:38	$10^{-14}$	13
70	100	23	16:11:19	00:01:03	$10^{-11}$	845
70	100	23	16:11:19	00:00:40*	$10^{-11}$	1440
10	100	112	308:53:46	00:01:50	$10^{-8}$	6836
1	100	152	-	00:02:59	-	-
1000	100	2	00:08:15	00:00:38	$10^{-14}$	13
700	100	3	00:18:19	00:00:38	$10^{-14}$	29
500	100	4	00:30:44	00:00:40	$10^{-14}$	46
1000	130	2	00:23:18	00:01:48	$10^{-14}$	13
2000	130	1	00:07:13	00:01:47	0	4
3000	130	1	00:07:26	00:01:51	0	4
4000	130	1	00:07:12	00:01:45	0	4

\* - помечаются то время, которое получается при использовании радиальной развертки

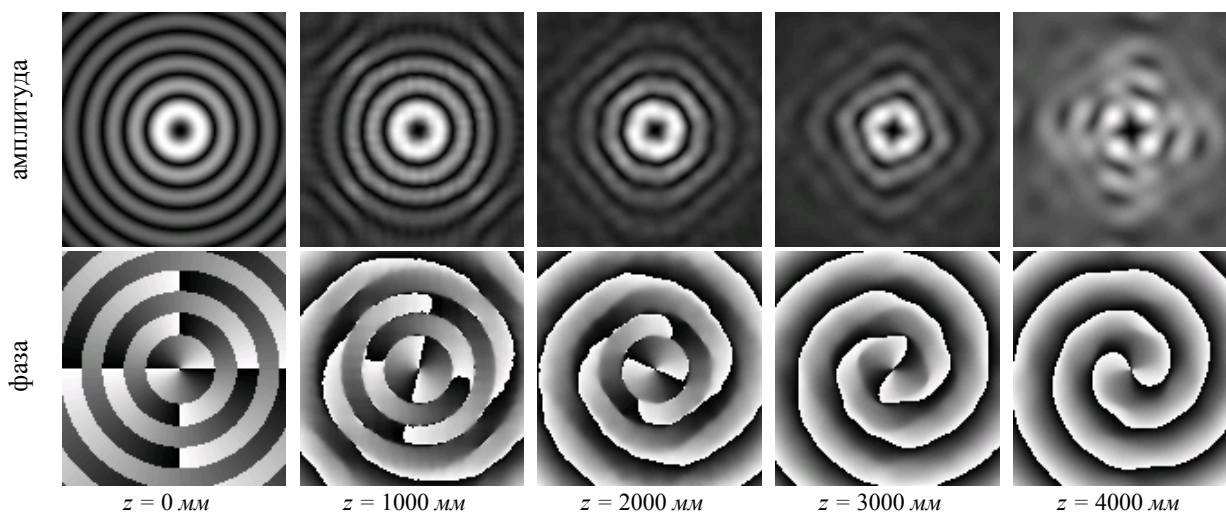


Рис. 7. Комплексное распределение моды Бесселя с параметрами:  $t=2$ ,  $\alpha=7$ ,  $C_{t,\alpha}=1$  на различных расстояниях от экрана

### ***Заключение***

В данной работе разработан быстрый алгоритм вычисления непараксиального интегрального оператора распространения лазерных полей в свободном пространстве. Алгоритм основан на использовании радиальной симметрии ядра интегрального преобразования и табулировании его значений. При этом повышение точности вычисления интеграла не за счет увеличения числа отчетов входной функции, а разбиении их на дополнительные отчеты для более детального представления быстро осциллирующего ядра преобразования, что существенно снижает вычислительную нагрузку.

При реализации всех предложенных способов оптимизации алгоритма удалось получить значительное ускорение его работы, благодаря чему для отдельно взятой задачи результат можно получить не за неделю работы программы, а за считанные минуты (см. табл. 1) притом, что по-

грешность будет близка к погрешности машинных вычислений.

### ***Благодарности***

Работа выполнена при поддержке российско-американской программы «Фундаментальные исследования и высшее образование» (CRDF Project SA-014-02), гранта РФФИ 05-01-96505, а также гранта INTAS 04-77-7198.

### ***Литература***

1. Виноградова М.Б., Руденко О.В., Сухоруков А.П. Теория волн. – М.: Наука, 1979. – 383 с.
2. Бахвалов Н.С. Численные методы (анализ, алгебра, обыкновенные дифференциальные уравнения). – М.: Наука, 1975. – 630 с.
3. Дроздов М.А. и Хонина С.Н. Исследование границ применимости параксиального приближения для описания распространения лазерного света в свободном пространстве // Естественные науки, экономика, менеджмент, 2003. № 4. С. 47-55.