

УДК 621.382:621.395

СРЕДСТВА ОТОБРАЖЕНИЯ 3D-ГРАФИКОВ ФУНКЦИЙ В РАЗЛИЧНЫХ СИСТЕМАХ КООРДИНАТ

К.П. КУРЕЙЧИК, С.Л. КУЧИНСКИЙ, В.А. ХАРЛАНОВ

*Белорусский государственный университет информатики и радиоэлектроники
П. Бровка, 6, Минск, 220013, Беларусь*

Поступила в редакцию 17 октября 2003

В данной статье приводится описание программного средства, предназначенного для отображения различных функциональных зависимостей вида $z = f(x, y)$ в различных системах координат. Рассмотрены общие вопросы организации программы, методы проецирования трехмерных координат на плоскость, способы построения перспективных проекций, различные алгоритмы визуализации. Материал, излагаемый в данной статье, может быть полезен как школьникам, студентам и аспирантам в качестве пособия по основам компьютерной графики, так и разработчикам аналогичных решений в качестве одного из возможных вариантов решения данной задачи.

Ключевые слова: функция, график, интерфейс, отображение, структура.

Введение

Одной из задач математики является задача анализа функциональных зависимостей. Существует множество методов анализа функций вида $y=f(x)$, вместе с тем теме изучения поведения более сложных функциональных зависимостей уделяется гораздо меньшее внимание. Данный факт можно объяснить в том числе и затруднениями при графической интерпретации функций вида $z=f(x, y)$, представляющих собой различного рода поверхности. Таким образом, возникает необходимость в разработке программного средства, позволяющего строить графики функциональных зависимостей от двух переменных в различных системах координат и обладающего удобным пользовательским интерфейсом.

Данная программа должна выступать в качестве вспомогательного средства при анализе различных функциональных зависимостей, давать пользователю представления о поведении исследуемых им функций, закреплять умения пользователя анализировать функции по их графикам и определять вид функции по заданному аналитическому выражению (формуле), развивать представления о различных системах координат и принципах их организации.

Структура программы

В качестве языка программирования был выбран Microsoft Visual C++ 6.0. Структура программы приведена на рис. 1.

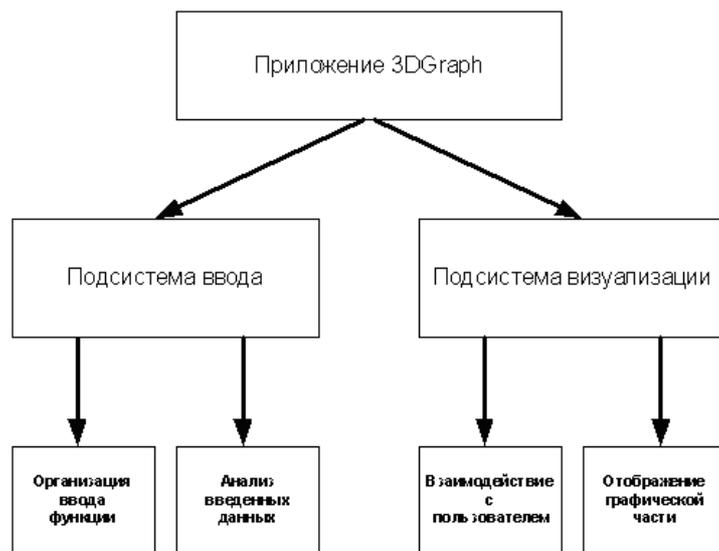


Рис. 1. Структура приложения

Логически разработанную программу можно представить в качестве целостной системы, которую можно разделить на две подсистемы: подсистема ввода и подсистема визуализации.

Подсистема ввода

Приложение 3Dgraph было включено в состав пакета STAR, который предназначен для проведения математических и статистических расчетов в различных областях науки, техники и производства.

В качестве подсистемы ввода приложения 3Dgraph выступает модуль математических расчетов Function. Внешний вид данного модуля приведен на рис. 2.

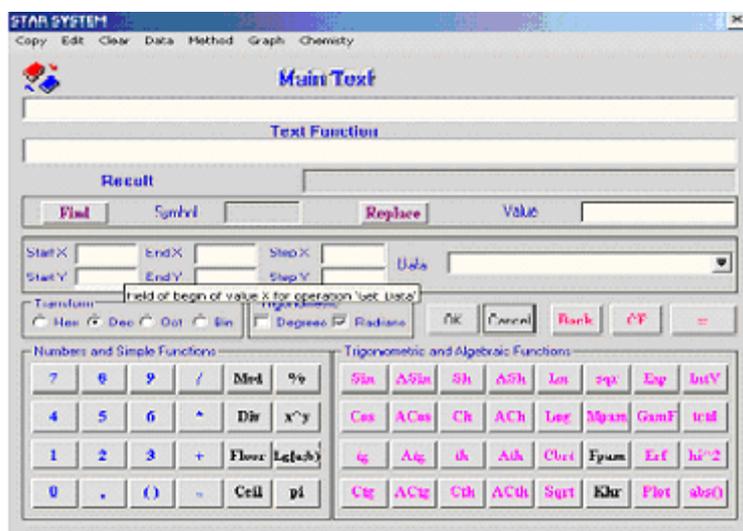


Рис. 2. Модуль Function

Модуль обеспечивает возможность задания пользователем функции в виде строки, указания символов, являющихся в данном выражении аргументами, задания границ и шага изменения каждого из аргументов. После чего производится интерпретация введенного выражения и вычисление всех значений функции при заданных значениях аргументов. Кроме вычисления значений функции, модуль Function может выполнять функции универсального калькулятора с широкими вычислительными возможностями (вычисление F критерия, критерия Кохрена, гам-

ма-функции, функции ошибок, плотности вероятности, интеграла вероятности, коэффициента Стьюдента, критерия χ^2 – квадрат и др.).

Подсистема визуализации

В качестве входных данных для подсистемы визуализации служит результирующий массив трехмерных координат, полученный в результате обработки подсистемы ввода, т.е. подсистема визуализации является функционально независимой по отношению к подсистеме ввода, что делает возможным ее автономное применение. В этом заключается одно из ее достоинств, так как это позволяет применять в качестве источника данных произвольный генератор координат точек (например, реальный физический прибор), что расширяет области применения данной подсистемы и не ограничивает ее только теми возможностями, которые были поставлены в качестве основных задач при создании программы.

В подсистеме визуализации реализованы два основных класса, один из которых отвечает непосредственно за отображение графика функции, а второй предназначен для организации взаимодействия с пользователем.

Рассмотрение работы подсистемы начнем с общего ознакомления с интерфейсом. Итак, после завершения этапа вычисления производится вывод на экран окна (рис.3), большая часть которого отведена непосредственно под отображаемую функцию. Окно снабжено небольшим меню. Непосредственно под меню располагаются элементы управления, позволяющие модифицировать полученный график. Ближе к левому краю располагается переключатель стиля отображения графика. Ближе к правому краю находится небольшая панель, отвечающая за цветовую окраску отображаемого графика и фона, на котором происходит отображение данного графика.

Взаимодействие с пользователем организовано посредством реализации следующих сервисных функций:

- сохранение полученного графика в файл;
- загрузка ранее сохраненного графика из файла;
- изменение цветовой раскраски графика и фона;
- возможность изменения стиля отображения графика;
- вращение графика в двух плоскостях;

Сохранение полученного графика в файл. После получения функциональной зависимости у пользователя может появиться желание сохранить результаты своей работы. Именно с этой целью реализована возможность сохранения текущего графика в текстовый файл. Выбор данной функции реализован через меню. При активизации данной функции происходит сохранение текущего массива точек в файл. Помимо этого в файл сохраняется количество отсчетов по каждой переменной и признак текущей системы координат.

Формат файла следующий. В первую строку занесен признак системы координат. Он равен 0 для декартовой, 1 для цилиндрической, 2 для сферической. Вторая и третья строки – число отсчетов для первого и второго аргументов функции. Далее идут строки, содержащие значения X, Y и Z координат точек, принадлежащих графику.

Загрузка ранее сохраненного графика из файла. Позволяет пользователю произвести загрузку ранее сохраненного графика и продолжить с ним работу. Выбор данной функции также реализуется через меню. При выборе пункта Load устанавливается соответствующий тип системы координат, а массив точек инициализируется координатами из файла.

Изменение цветовой раскраски графика и фона. При помощи данных функций, пользователь может изменить установленные по умолчанию значения цветов для сетки графика и фона. В правом верхнем углу окна находится элемент управления, состоящий из двух статических элементов, демонстрирующих текущий цвет линии и фона. При указании курсором мыши на один из элементов и нажатии левой кнопки на экране появляется диалог выбора цвета. По завершении выбора цвета пользователем происходит переустановка соответствующего цвета.

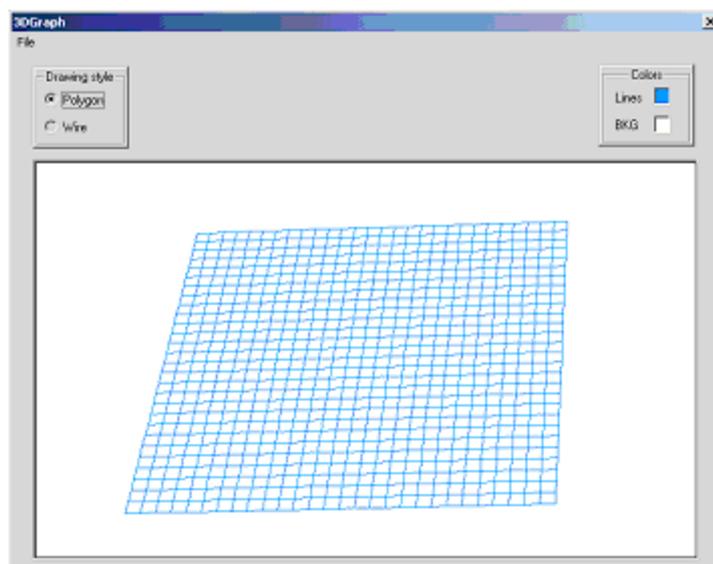


Рис. 3. Подсистема визуализации

Возможность изменения стиля отображения графика. В программе предусмотрена возможность отображения графика функций в двух режимах. Выбор режима осуществляется путем установки переключателя, расположенного в левом верхнем углу окна. Первый режим (Wire) — производится отображение графика в виде проволочного каркаса. Второй режим (Polygon) — производится отображение графика при помощи полигонов. Первый режим обеспечивает большую скорость отображения, но значительно меньшую наглядность получаемого изображения. Второй же режим, наоборот, отображение более медленное, зато получаемое изображение получается весьма наглядным и легко понимаемым. На рис. 4 изображен график функции в режиме Wire, а на рис. 5 тот же самый график в режиме Polygon

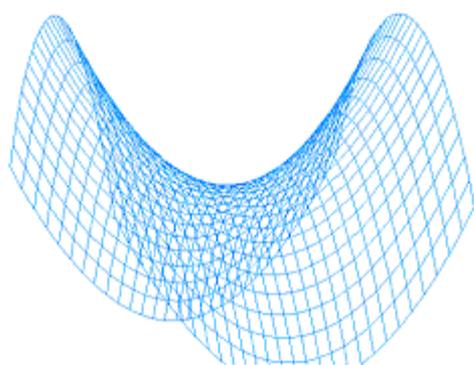


Рис. 4. Режим Wire

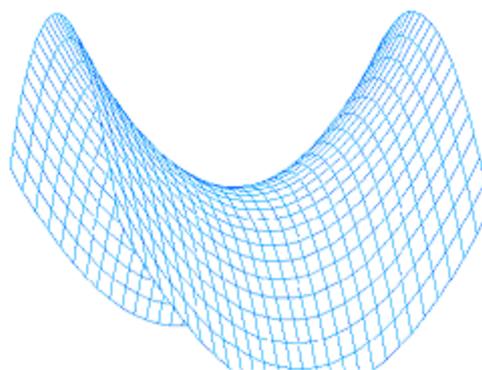


Рис. 5. Режим Polygon

Вращение графика в двух плоскостях. Одним из главных достоинств программы является возможность вращения полученного графика, что позволяет изучить построенный им график более детально. Это реализуется при помощи мыши. Если пользователю необходимо осуществить поворот графика, то надо в области отображения графика нажать левую кнопку мыши и, продолжая удерживать ее нажатой, переместить курсор мыши в том направлении, в котором необходимо повернуть график. При отпускании кнопки вращение прекращается. На рис. 6 изображен график функции в начальный момент работы приложения 3Dgraph (т.е. углы поворота графика относительно пользователя равны нулю). На рис. 7 изображен тот же график после поворота его пользователем при помощи мыши.



Рис. 6. Начало работы

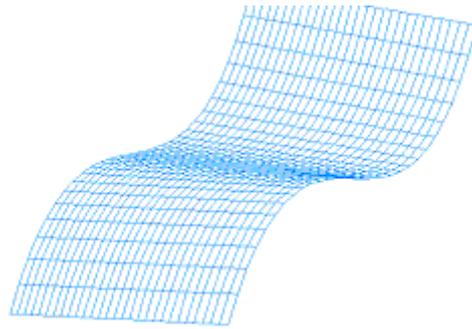


Рис. 7. После поворота

Класс, отвечающий за отображение графика функции, реализует непосредственный вывод графика функции на экран средствами GDI (Graphics Device Interface), а также осуществляет ряд сопутствующих математических преобразований:

- вычисление коэффициента сжатия изображения;
- преобразование трехмерных координат точек в экранные (двумерные);
- преобразование массива трехмерных координат точек в массив полигонов;
- отображение графика в виде проволочного каркаса;
- отображение графика полигонами.

Вычисление коэффициента сжатия изображения. Для того чтобы отображаемый график помещался в отведенную ему область, необходимо все координаты каждой точки умножить на соответствующий корректирующий коэффициент. Данные коэффициенты (индивидуальные для каждой координатной оси) должны учитывать как размеры области для отображения, так и максимальные и минимальные значения точек по соответствующим осям.

Преобразование трехмерных координат точек в экранные точки. Функция производит преобразование точек, хранящихся в массиве трехмерных точек (полученном в результате отработки подсистемы ввода), в соответствующее им двумерное представление. При этом преобразование производится с учетом текущих углов поворота графика относительно пользователя, а также с учетом перспективы (для еще большего повышения наглядности). Вычисление координат точек производится по следующим формулам:

$$M = \frac{1}{1 - \frac{x \sin\theta \cos\varphi + y \sin\theta \sin\varphi + z \cos\theta}{L}}$$

$$x' = (-x \sin\varphi + y \cos\varphi) M,$$

$$y' = (-x \cos\theta \cos\varphi - y \cos\theta \sin\varphi + z \sin\theta) M,$$

где x, y, z — координаты текущей точки в пространстве; x', y' — экранные координаты текущей точки; φ, θ — углы поворота графика относительно осей x и y соответственно; M — масштабный множитель; L — расстояние от зрителя до сцены.

Преобразование трехмерных координат точек в экранные полигоны. Данная функция применяется при отображении графика полигонами. Массив трехмерных точек, принадлежащих графику, преобразуется во временный массив полигонов, каждый элемент которого содержит четыре координаты, соответствующие вершинам полигона, и расстояние до полигона. Преобразование вершин полигонов производится по формулам, аналогичным приведенным выше.

Отображение графика в виде проволочного каркаса. При отображении графика в виде проволочного каркаса производится вывод точек, соединенных линиями, из массива точек, полученного в функции преобразования трехмерных координат точек, на экран.

Отображение графика полигонами. При отображении графика с помощью полигонов производится вывод полигонов из временного массива двумерных точек, полученного в функ-

ции преобразования массива трехмерных координат точек в массив полигонов, на экран. При этом в качестве алгоритма отображения полигонов был выбран "алгоритм художника", как один из наиболее простых, достаточно производительных и эффективных.

Заключение

В заключение хочется отметить, что для максимального соответствия потребностям пользователя при разработке интерфейса программы был учтен ряд факторов психолого-педагогического характера.

Информативность. Программа выдает пользователю подробную информацию как в графической, так и в текстовой (в виде файла) форме.

Наглядность. Программа демонстрирует графики с учетом особенностей зрительного восприятия человека (перспективного проецирования).

Динамичность. Предусмотрена возможность вращения графика.

Возможность варьирования пользователем режимов работы программы. Предусмотрена возможность выбора режима изображения графиков с отсечением невидимых поверхностей графика и без их отсечения.

Простота управления работой модели. Интерфейс модели интуитивно понятен и удобен пользователю.

Кроме всего прочего программа может быть полезна для визуализации данных при компьютерном моделировании трехмерных задач, так как она значительно повышает наглядность представления результатов моделируемого процесса, делает его интерпретацию более унифицированной.

METHODS OF MATH FUNCTIONS 3D-DIAGRAMS VISUALISATION IN VARIOUS COORDINATE SYSTEMS

K.P. KUREJCHIK, S.L. KUCHINSKIY, V.A. HARLANOV

Abstract

Article results the description of the software intended for display of various functional dependences of a kind $z = f(x, y)$ in various systems of coordinates. Article considers common questions of the program organization, methods of displaying of three-dimensional coordinates on a plane, ways of construction of perspective projections, various algorithms of visualization. The material stated in the article may be useful as to schoolboys, students and post-graduate students as the manual for bases of computer graphics, as to developers of similar decisions as one of possible decisions on the given problem.

Литература

1. Роджерс Д., Адамс Дж. Математические основы машинной графики: Пер. с англ. М., 1980.
2. Слободянюк А.И., Титовицкая А.Э. Машинная графика: линии и поверхности. Мн., 2000.
3. Томпсон Н. Секреты программирования трехмерной графики для Windows 95: Пер. с англ. СПб., 1997.
4. Грегори К. Использование Visual C++ 6. Специальное издание: Пер. с англ. М.; СПб.; К., 2000.