

Разработка проблемно-ориентированных систем геометрического моделирования

Карабчевский В.В., Хлепитько И.В.
кафедра ПМИИ ДонНТУ
karabch@pmi.dgtu.donetsk.ua

Abstract

Karabchevsky V., Hlepitko I. Development of geometrical modeling problem-orientated systems. The tools of development of geometrical modeling problem-orientated systems and the results of their application are described.

Введение

В настоящее время одной из самых широко известных и часто используемых систем автоматизации проектирования является система AutoCAD. Она предоставляет пользователю широкий спектр функциональных возможностей для работы с чертежами, но и стоимость лицензионной копии этой программы довольно велика. Целью данной статьи является описание разработки бесплатного графического векторного редактора, предоставляющего пользователю подмножество функций системы AutoCAD LT.

В [1] обоснована целесообразность разработки независимого от AutoCAD или других распространенных графических систем программного обеспечения, предназначенного для моделирования геометрических фигур, в [1,2] описаны различные программные реализации этого подхода и результаты их использования в учебном процессе. В [3] предлагается использование свободно распространяемого программного обеспечения (LinuxCAD, QCAD), авторы допускают также возможность самостоятельных разработок, не забывая при этом о том, что средства разработки (языки, библиотеки) могут являться лицензионными программными продуктами. В [4] описан состав графического ядра, представляющего минимальный набор средств, необходимых для разработки векторных двумерных и трехмерных систем геометрического моделирования, применение этих средств для создания графической системы, повышающей наглядность преподавания начертательной геометрии. В этой же работе предлагается дальнейшее развитие графического ядра с целью использования его в качестве средства для разработки легких систем геометрического моделирования, которые могут применяться как в учебных, так и в практических целях, реализации части описанных положений посвящена статья [5]. Ниже описана новая реализация такого ядра и результаты ее использования.

Принципы функционирования программы и хранения данных

В программе используется два слоя изображений. На нижнем слое отображаются уже созданные объекты, с которыми на данный момент пользователь не производит никаких действий. Верхний слой является прозрачным относительно нижнего. На него помещаются объекты в момент их создания, когда заданы не все данные об объекте и его вид часто изменяется в зависимости от текущего положения курсора. Кроме того, на него перемещаются объекты, которые модифицируются.

В каждый момент времени программа может находиться в одном из нескольких состояний: ожидание, построение одного из базовых объектов, поворот объектов, перенос объектов, зуммирование, панорамирование. В зависимости от текущего состояния программа будет определенным образом реагировать на действия пользователя.

Все объекты хранятся в списках, которые являются атрибутами для класса TGrObj. Рассмотрим структуру этого класса. Поля LLine, LCircle и LEllipse – списки линий, окружностей и эллипсов соответственно. Элементами списка являются объекты классов TLine, TCircle и TEllipse соответственно. Поля Line, Circle, Ellipse используются для временного хранения данных об объектах в процессе создания объектов. Метод ReDraw перерисовывает все созданные пользователем объекты.

Объекты классов TLine, TCircle и TEllipse используются для хранения информации о базовых объектах. Каждый класс имеет следующие поля: Num (номер объекта), Style (стиль линии, определяющей объект), Color (цвет линии, определяющей объект), Width (толщина линии, определяющей объект) и Select (флаг, указывающий является ли объект выбранным). Кроме того, в каждом из рассмотренных классов есть поля, в которых хранится информация, определяющая объект на плоскости. Для отрезков это начальная и конечная точки, для окружностей – центр и радиус, для эллипсов – центр, длины полуосей и угол наклона эллипса относительно горизонтальной оси. Также в каждом классе имеются методы, которые отвечают за добавление объекта класса в список объектов (Add), восстановление системы в режим ожидания при отмене пользователем команды рисования (Cancel), перерисовку объекта (Draw) и прорисовку объекта как выделенного (Selected). Интерфейс системы представляет собой подмножество интерфейса AutoCAD-2008 LT и позволяет обеспечивать все необходимое множество функций и достаточно комфортную работу пользователя, имеющего опыт работы в AutoCAD. На рисунке 1 приведен вид окна системы.

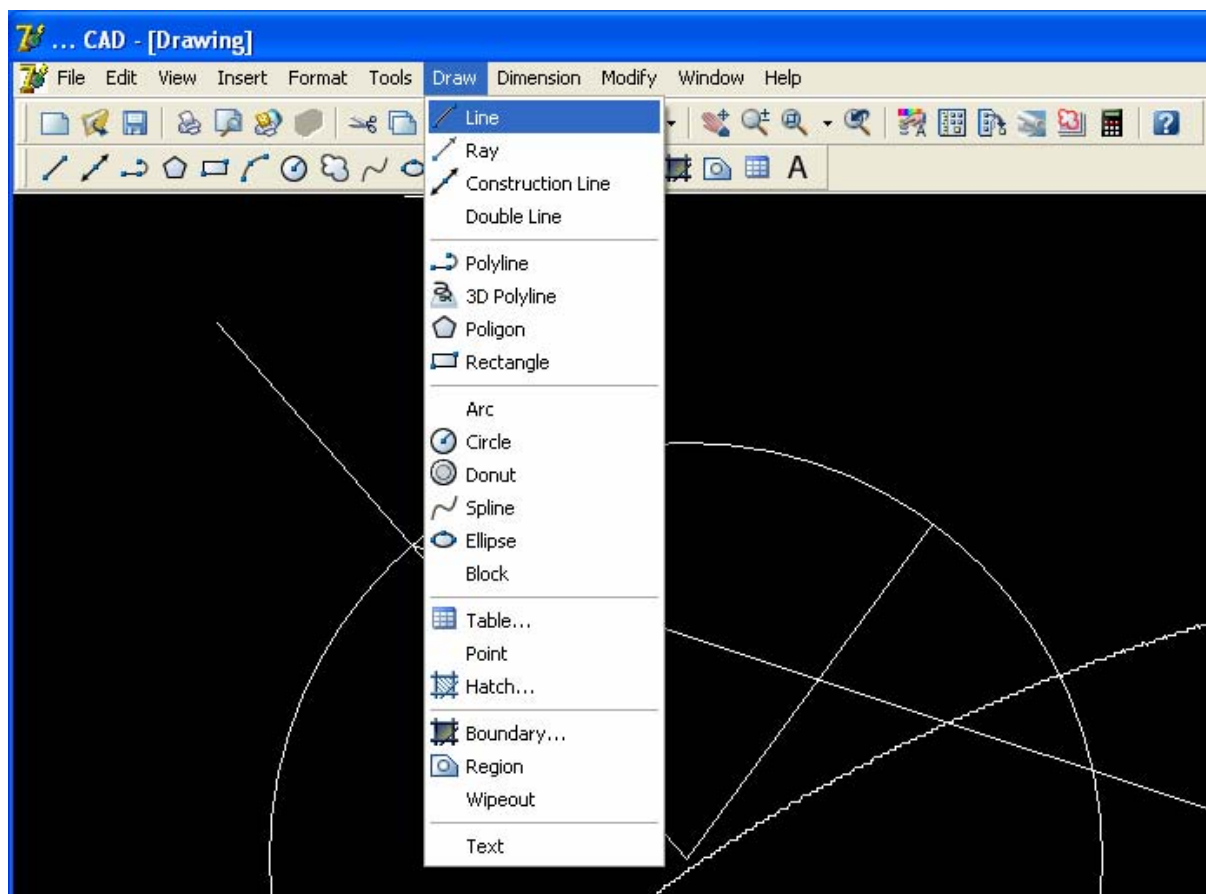


Рисунок 1 – Пример работы в системе

Система обеспечивает значительную часть методов построения двумерных объектов AutoCAD. В качестве примера рассмотрим построение окружностей. Пользователь может задать окружность тремя способами: указать центр и радиус окружности, указать отрезок, являющийся диаметром окружности или указать три точки, принадлежащие окружности. В первом случае данные сразу заносятся в структуру, так как известны центр и радиус окружности. Во втором случае вычисляется середина отрезка, эта точка является радиусом окружности, а радиусом является половина расстояния между концами отрезка. Рассмотрим подробнее случай, когда окружность задана тремя точками. В этом случае необходимо найти окружность, описанную вокруг треугольника, заданного этими точками. Для этого нужно найти точку пересечения срединных перпендикуляров этого треугольника. Можно определить уравнения срединных перпендикуляров, зная, что они проходят через среднюю точку грани треугольника и что нормальный вектор для этой грани является направляющим для перпендикуляра. Решив систему для двух перпендикуляров, найдем точку их пересечения. Координаты этой точки показаны на рисунке 2.

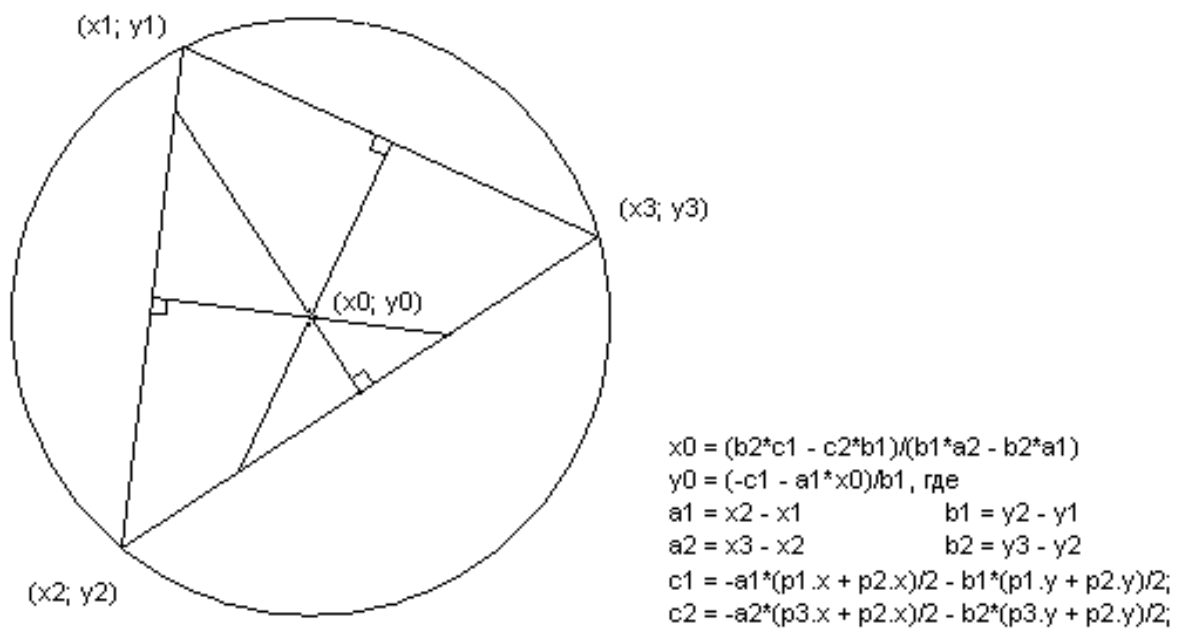


Рисунок 2 – Построение окружности по трем точкам

Модификация объектов

Перенос объектов может осуществляться двумя способами. В первом случае в состоянии «перенос объектов» система переходит из состояния ожидания, после того как пользователем были выбраны некоторые объекты и хотя бы для одного из них был сделан клик мышью на точке переноса. Для отрезков такой точкой является средняя точка, для окружностей и эллипсов – центр. Таким образом, при получении сообщения о нажатии левой кнопки мыши в состоянии ожидания происходит перебор всех выделенных объектов. Для каждого из них определяется точка переноса и проверяется, принадлежат ли координаты курсора некоторой области вокруг этой точки. Выбранные таким образом элементы помещаются в специальный стек, после чего происходит переход в состояние «перенос объектов». Далее система работает только с объектами из этого стека. При перемещении курсора перемещаются и объекты.

Во втором случае переход в состояние «перенос объектов» осуществляется путем задания пользователем команды «move». В этом случае система проверяет наличие элементов в списке выбранных объектов. Если в список выбранных объектов пуст, то система предлагает пользователю выбрать объекты, которые нужно переместить. Иначе система работает с уже выбранными объектами. При движении мыши положение перемещаемых объектов не изменяется, показывается лишь

возможное положение. Только после выбора пользователем конкретной точки объекты будут перемещены.

При повороте объектов выполняется практически та же последовательность действий, что и при переносе, за тем лишь исключением, что для каждого объекта определяющие его точки не переносятся, а поворачиваются. В этом случае в обработчике события движения мыши через текущую точку и точку, относительно которой будет происходить поворот объектов, проводится линия, вычисляется ее угол наклона относительно горизонтальной оси и на экран выводится изображение объектов, повернутых относительно своего начального положения на этот угол.

Реализация привязок

При создании объектов пользователь может использовать уже имеющиеся данные. При нажатой кнопке OSNAP работает режим привязок. В этом режиме при создании объекта программа просматривает все объекты и в зависимости от параметров привязок определяет конечные точки и середины отрезков, центры окружностей и эллипсов, точки пересечения отрезков между собой, отрезков с окружностями и эллипсами и окружностей между собой. Если расстояние от курсора мыши до одной из таких точек меньше фиксированного значения, то курсор перемещается в эту точку. Если таких точек несколько, то курсор помещается в ближайшую из них. Для отыскания точек пересечения необходимо осуществлять полный перебор объектов и проверять каждую пару. Найти точки пересечения объектов можно решением системы из двух уравнений, каждое из которых характеризует один из пары рассматриваемых объектов. При поиске ближайшей точки необходимо запоминать также код привязки.

Однако в случае пересечения эллипсов между собой или эллипса с окружностью такой подход не дает желаемого результата. С аналитическим решением такой системы уравнений не справляются даже такие пакеты как Mathematica и MathCAD. В этом случае следует пользоваться различными приближенными методами. Например, эллипс можно рассматривать как полилинию, как это делалось в ранних версиях AutoCADa. Однако в данном случае для высокой точности вычислений необходима высокая степень аппроксимации. Еще одним вариантом решения проблемы является аппроксимация эллипса дугами. Это даст большую точность при низкой степени аппроксимации, но затрудняет процесс аппроксимации. Самым независимым от условий пересечения объектов способом является применение численных методов решения систем уравнений.

Выводы и перспективы исследований

Применение разработанной системы в учебном процессе для решения задач начертательной геометрии позволяет одновременно освоить основы AutoCAD, что актуально в условиях ограниченного количества лицензионных копий этой системы. Возможность сохранения и загрузки чертежей в формате dxf обеспечивает совместимость с AutoCAD и дает возможность использования системы для решения ряда практических задач, состав которых будет расширяться в ходе продолжения разработки.

Литература

1. Карабчевский В.В. Повышение качества преподавания инженерной графики путем разработки и применения обучающих систем. // Научные труды Донецкого государственного технического университета. Серия: Информатика, кибернетика и вычислительная техника, (ИКВТ-99) выпуск 6: – Донецк: ДонГТУ, 1999. – С. 294-299.
2. Карабчевський В.В. Засоби розробки навчальних систем для курсу “Інженерна графіка” // Прикладна геометрія та інженерна графіка. Міжвідомчий науково-технічний збірник. Випуск 71. Київ: КНУБА. – 2002. – С. 195-200.
3. Салтавец В.І., Печерцев О.О., Герасименко В.В., Салтавец М.В. Створення програмного забезпечення для виконання графічних робіт // Сборник трудов 7-й Международной научно-практической конференции «Современные проблемы геометрического моделирования». Мелитополь: ТГАТА. – 2003. – С. 44-48.
4. Карабчевський В.В. Засоби зв'язку між операціями над двовимірними і тривимірними моделями. // Наукові праці Донецького національного технічного університету. Серія Інформатика, кібернетика та обчислювальна техніка, випуск 93. Донецьк: ДонНТУ. – 2005. – С. 41- 46.
5. Карабчевский В.В., Хлепотько И.В. Графический редактор для работы с параметризованными двумерными объектами // Матеріали ІІІ наукової конференції молодих учених та студентів «Інформатика та комп'ютерні технології». Донецьк: ДонНТУ. – 2007. – С. 278-280.

Дата надходження до редакції 29.12.2007 р.