

Санкт-Петербургский государственный университет

Направление Математическое обеспечение и администрирование
информационных систем

Профиль Математические основы информатики

Винникова Алёна Андреевна

Редактор параметрической двумерной
геометрии для программного комплекса
ELCUT

Бакалаврская работа

Научный руководитель:
старший преподаватель Симуни М. Л.

Рецензент:
ведущий программист ООО "ТОР" Блюдзе М. Ю.

Санкт-Петербург
2016

SAINT-PETERSBURG STATE UNIVERSITY

Mathematical Software and Information Systems Administration
Mathematical foundations of computer science

Alyona Vinnikova

Editor of two-dimensional parametric
geometry for ELCUT program complex

Bachelor's Thesis

Scientific supervisor:
Senior lecturer Simuni M. L.

Reviewer:
Senior software engineer OOO "TOR" Bliudze M. Y.

Saint-Petersburg
2016

Оглавление

Введение	5
1. Цели и задачи работы	7
2. Причины возникновения проблемы и её актуальность	8
2.1. Понятие параметрического моделирования	8
2.1.1. Системы инженерного проектирования и моделирования	8
2.1.2. Параметризация и её преимущества	9
2.2. Программный комплекс ELCUT, основные возможности	10
2.3. Обзор существующих подходов к параметризации	12
2.3.1. AutoCAD	14
2.3.2. Pro/ENGINEER	15
2.3.3. КОМПАС	15
2.3.4. SolidWorks	16
2.4. Актуальность задачи создания параметрического редактора для ПК ELCUT	18
3. Проектирование системы	19
3.1. Выбор используемых технологий	19
3.2. Проектирование функциональности	19
3.2.1. Геометрические примитивы	20
3.2.2. Понятие фигуры	20
3.2.3. Возможности редактирования	21
3.2.4. Параметризация	21
3.2.5. Проверка совместимости	24
3.2.6. Преобразования	24
3.3. Проектирование пользовательского интерфейса	25
4. Реализация	26
4.1. Описание пользовательского интерфейса	26
4.1.1. Вкладка "Элементы"	26

4.1.2. Вкладка "Параметры"	28
4.1.3. Вкладка "Преобразования"	31
4.2. Описание реализации	31
4.3. Анализ результатов тестирования и опытной эксплуатации	33
4.4. Основные результаты	33
Заключение	35
Список литературы	36

Введение

В современном мире компьютерные технологии получают всё более и более широкое распространение. Повсеместное использование получает автоматизация инженерного проектирования и моделирования. Действительно, с помощью машинных вычислений можно сильно сократить затраты, например, на время, использование интеллектуальных ресурсов и материалов. Системы автоматизированного проектирования и моделирования (САПР или CAD/CAE [18]) стали привычным инструментом любого конструктора.

Системы CAD/CAE включают в себя множество программных и аппаратных средств — от систем двумерного черчения до трехмерного параметрического моделирования поверхностей и объемных тел [27]. Существует большое количество удобных систем моделирования и проектирования, такие как AutoCAD [23], FreeCAD [5], Компас [19] и т.д. С помощью систем проектирования можно создавать специализированные трёхмерные и двумерные чертежи и получать визуализацию моделей. Системы моделирования позволяют понять, какими свойствами будет обладать модель в реальных условиях эксплуатации.

В данной бакалаврской работе реализован прототип графического редактора, обеспечивающего возможность параметризации объектов. Реализация редакторов такого типа, в общем случае, является довольно сложной, но очень важной и актуальной конструкторской задачей — задачей параметрического моделирования [8]. Параметрическое моделирование лежит в основе многих систем автоматизированного проектирования. Параметризация позволяет за короткий промежуток времени рассмотреть различные конструктивные схемы и избежать принципиальных ошибок.

Одной из распространенных отечественных систем моделирования является программный комплекс ELCUT [4]. В настоящее время в рамках разработки новой версии программы ELCUT появилась необходимость в новом геометрическом редакторе с возможностью параметризации. При его разработке возникла необходимость в исследовании

различных вариантов интерфейса и способов реализации. Программа, реализованная в данной бакалаврской работе, демонстрирует один из таких способов для двумерной геометрии.

1. Цели и задачи работы

Целью данной работы является обеспечение возможности параметризации для графического редактора двумерной геометрии. С помощью этого редактора можно создавать объекты (как простые, например, точки и отрезки, так и более сложные — окружности и многоугольники) и фигуры, состоящие из совокупности объектов. При объединении объектов задаются отношения между ними, то есть фиксируются некие параметры. Параметризация [21] позволяет автоматически изменять объекты.

Для выполнения этой цели были поставлены следующие задачи:

- изучить основные возможности существующих систем параметризации;
- рассмотреть особенности ELCUT и его требований к параметризации;
- выбрать набор функциональности для реализации;
- спроектировать интерфейс и разработать архитектуру редактора;
- реализовать программу;
- проанализировать результаты проделанной работы с точки зрения удобства и набора возможностей.

2. Причины возникновения проблемы и её актуальность

2.1. Понятие параметрического моделирования

2.1.1. Системы инженерного проектирования и моделирования

Сегодня невозможно представить процесс проектирования без использования компьютерных технологий. Создание чертежей всегда важный и трудоемкий процесс, который раньше требовал достаточно кропотливой работы. Это целая наука, которая формировалась и модернизировалась многие годы вместе с инженерным процессом. С такой работой мог справиться не каждый человек — малейшая оплошность и вся работа насмарку. При любых изменениях в планах проекта приходилось заново создавать чертежи, что отнимало много времени и сил.

С появлением компьютеров такие проблемы стали не актуальны, так как начали появляться и повсеместно использоваться различные системы инженерного проектирования. Такие системы обозначают аббревиатурой CAD [2] от английского Computer-Aided Design, что приблизительно означает «проектирование при помощи компьютера». Они значительно упростили работу инженеров, избавив их от некоторых рутинных занятий, возникавших при минимальной модификации чертежей.

Автоматизированные системы проектирования позволяют:

- сократить трудоёмкость проектирования и планирования;
- сократить сроки проектирования;
- сократить себестоимость проектирования;
- повысить уровень качества результатов проектирования.

Итак, система CAD — это некоторый набор технических и программных средств, реализующий информационную технологию выполнения функций проектирования. Наиболее известными программными

продуктами являются AutoCAD и FreeCAD. Наряду с этим понятием, существует понятие CAE [3] системы (от английского Computer-Aided Engineering — ”автоматизированное конструирование”). Это различные программные продукты, позволяющие при помощи математических расчётных методов оценить, как поведёт себя компьютерная модель изделия в реальных условиях эксплуатации, то есть осуществляют динамическое моделирование, проверку и оптимизацию изделий и средств их производства. С помощью таких систем, например, можно проверить работоспособность изделия, без привлечения больших затрат времени и средств. Широко распространенной системой инженерных расчётов является SolidWorks [11], среди российских разработок стоит, конечно же, отметить ELCUT.

В работе CAE системы выделяют следующие этапы [14]:

- предварительная обработка данных и построение по геометрической модели объекта математической модели, которая задается множеством уравнений с различными переменными;
- анализ модели;
- обработка результатов — визуализация расчетов математической модели.

Современные CAE системы зачастую интегрируются в CAD системы, в этом случае получаются гибридные CAD/CAE. Для этого в русском языке есть термин САПР — система автоматизированного проектирования и моделирования.

2.1.2. Параметризация и её преимущества

Многие CAD/CAE системы базируются на координатном подходе [15]. То есть при создании, например, линии программа записывает и хранит координаты концевых точек каждой линии. Это не очень удобно. Если необходимо каким-либо образом изменять чертеж, требуется обновление вручную многочисленных линий, на которых сказалось данное изменение.

Вышеупомянутую проблему решает параметрический подход [22], который позволяет мыслить в терминах геометрических ограничений и размеров, без привязки к определенным координатам. В данном случае, создаются геометрические примитивы (точки, отрезки, окружности и так далее), для которых задаются размерности и геометрические ограничения (например, равенство длин, параллельность или перпендикулярность и прочее). Программа самостоятельно строит систему уравнений, в которой координаты точек будут неизвестными, а размеры и геометрические ограничения будут уравнениями, которые определяют конечный результат. При каких-либо изменениях не нужно вносить самостоятельно поправки в рисунок, система автоматически пересчитает координаты всех точек на основе заданных размеров и ограничений, после этого перерисует модель.

Параметрический метод значительно удобнее и совершеннее координатного подхода. Подробнее про параметризацию изложено в разделе 2.3.

2.2. Программный комплекс ELCUT, основные возможности

ELCUT — программный комплекс (ПК), разработанный российской компанией ООО "ТОР", предназначенный для инженерного моделирования электромагнитных, тепловых и механических задач методом конечных элементов [20]. ELCUT широко используется в научных исследованиях, промышленности и образовании. Данная САЕ система является популярной среди инженеров, преподавателей и студентов технических специальностей, таких как физика и электротехника.

Программа имеет дружественный пользовательский интерфейс, отличается простотой описания сложных моделей и широким спектром аналитических возможностей. При этом работать с данной программой удобно и легко, что позволяет сосредоточиться на решаемой задаче и не отвлекаться на изучение интерфейса и используемых математических

методов. Поэтому ПК ELCUT особенно популярен среди студентов.

Программный продукт включает в себя [26]:

- Редактор данных — позволяет задавать физические характеристики модели и свойства сред;
- Решатель — преобразует модели отдельных конечных элементов в общую систему алгебраических уравнений и рассчитывает её;
- Постпроцессор (анализ результатов) — служит для визуализации результатов решения в удобной для пользователя форме;
- Редактор геометрии — позволяет описать геометрию модели.

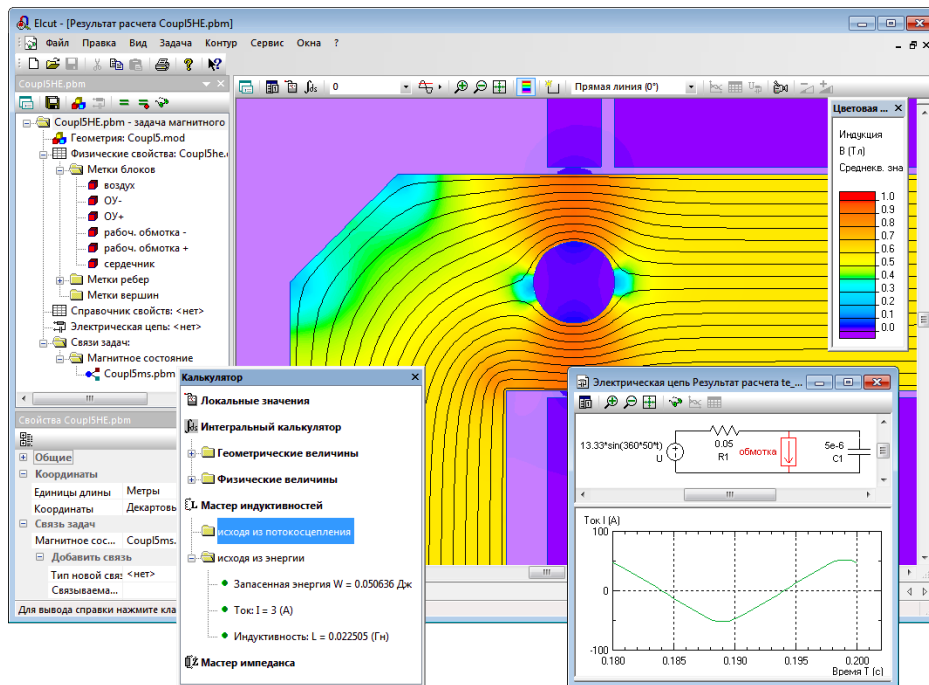


Рис. 1: Программный комплекс ELCUT

Важной особенностью ПК ELCUT является то, что он специализируется на решении двумерных (плоских) задач. То есть в текущей версии программы реализовано только двумерное моделирование. Стоит подчеркнуть, что на данный момент в геометрическом редакторе программы нет возможности параметризации моделей, что разработчики хотят исправить в следующей версии программы. Поэтому стала актуальной задача — разработать двумерный геометрический редактор с возможностью параметризации объектов.

2.3. Обзор существующих подходов к параметризации

В проектировании существует понятие ”геометрическое ядро” [28] — набор библиотек с программным интерфейсом, с помощью которого можно использовать функции геометрического моделирования, в том числе и параметрического моделирования. Многие современные системы инженерного проектирования и моделирования построены на основе собственных геометрических ядер, тогда как другие используют готовые лицензированные ядра. Существуют коммерческие ядра, например, Parasolid [9], и свободно распространяемые, например, ядро Open CASCADE [7].

Использование готовых ядер обусловлено несколькими причинами, главная из них: экономическая выгода. Геометрические ядра — это сложные и большие программы, над созданием которых могут трудиться не один год десятки, а то и сотни, квалифицированных инженеров и математиков. Заказать нужное геометрическое ядро у компании, специализирующейся на создании именно таких программ, дешевле, чем разрабатывать его самостоятельно.

Геометрическое ядро включает в себя [28]:

- описание формы моделируемого объекта
- описание связей элементов модели
- журнал построения модели
- атрибуты элементов модели

Как говорилось ранее, параметрическое проектирование — моделирование с использованием параметров элементов и соотношений между этими параметрами. Различают два основных вида параметрического проектирования — на базе предыстории [24] и вариационное проектирование [16].

Проектирование на базе предыстории (или иерархическое проектирование) отличается тем, что параметры, задающие новые элементы,

находятся в непосредственной зависимости от параметров элементов, построенных ранее. При изменении значения параметра конкретного элемента происходит пересчет формы всех элементов, параметры которых зависят от измененного. Таким образом эффективно реализуется схема обновления параметрической модели, однако ограничиваются выразительные возможности параметрического проектирования из-за запрета на циклические зависимости между параметрами. Параметризация на основе истории построений присутствует во всех САПР, использующих трёхмерное твердотельное параметрическое моделирование.

Другой разновидностью проектирования является вариационное проектирование. В данном случае связи между параметрами задаются с помощью некоторых декларативных конструкций, которые называются ограничения. Ограничения задают логические и параметрические связи между элементами (например, параллельность, касание, расстояние), не определяя конкретный способ пересчета параметров. Данная концепция выгодно отличается от моделирования на базе предыстории своими выразительными возможностями, но реализация схемы обновления модели при изменении параметров сложнее, так как требуется решение системы геометрических ограничений.

Различают несколько видов геометрических зависимостей: размерные и логические [1]. Размерные ограничения задают углы и расстояния, а логические — параллельность, перпендикулярность, касание и т.д.

Наиболее распространенными логическими ограничениями являются:

- совпадение объектов;
- касание;
- равенство длин;
- равенство радиусов;
- фиксация геометрии;

- фиксация длины и направления;
- точка на кривой;
- параллельность;
- перпендикулярность;
- горизонтальность;
- вертикальность;
- зеркальная симметрия.

Часто используют следующие размерные ограничения:

- расстояние;
- направленное расстояние;
- угол между объектами;
- радиус.

Далее рассмотрим несколько широко известных систем инженерного проектирования и моделирования и их основные возможности.

2.3.1. AutoCAD

В первую очередь хочется отметить AutoCAD [23]. Это САД система, выпущенная компанией Autodesk в 1982 году, предназначенная для подготовки технической документации, позволяющая строить чертежи практически любой сложности. AutoCAD активно используется в машиностроении, строительстве, архитектуре и других отраслях промышленности. С его помощью можно создавать элементарные фигуры, называемые "графические примитивы" (например, дуги, окружности, отрезки и т.д.) и редактировать их, создавая сложные чертежи. Редактирование объектов предусматривает возможность удаления, копирования, перемещения объектов, изменения их свойств (цвет, вес линий,

тип линий и др.), а также возможность выполнения операций удлинения, обрезания, зеркального отражения, поворота на заданный угол и т.д.

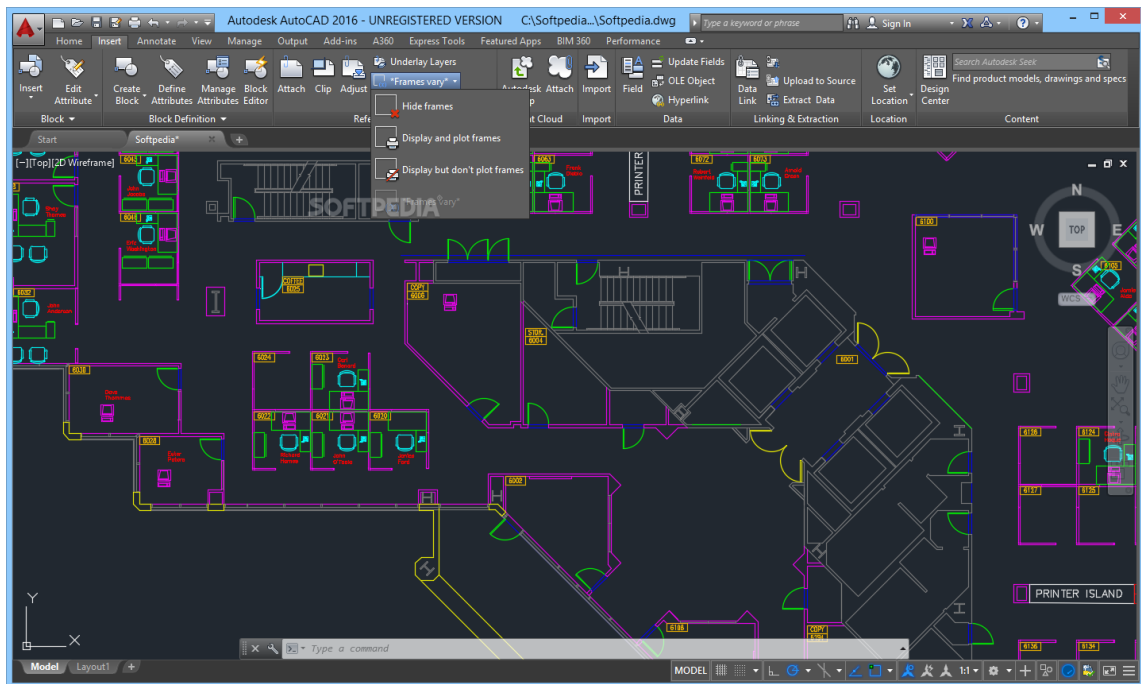


Рис. 2: AutoCAD

2.3.2. Pro/ENGINEER

Далее хотелось бы упомянуть такой программный продукт как Pro/ENGINEER [10], созданный американской корпорацией PARAMETRIC TECHNOLOGY CORPORATION (PTC), основанной русским эмигрантом Самуилом Гейсбергом в 1985 году. Геометрия Pro/ENGINEER основывается на геометрических примитивах — фичерах. Помимо данных о геометрии они содержат полную информацию о построении, полную историю создания объекта, связи с родительскими и подчиненными объектами.

2.3.3. КОМПАС

Из российских разработок популярной системой для конструирования в машиностроении и для проектирования в строительстве является

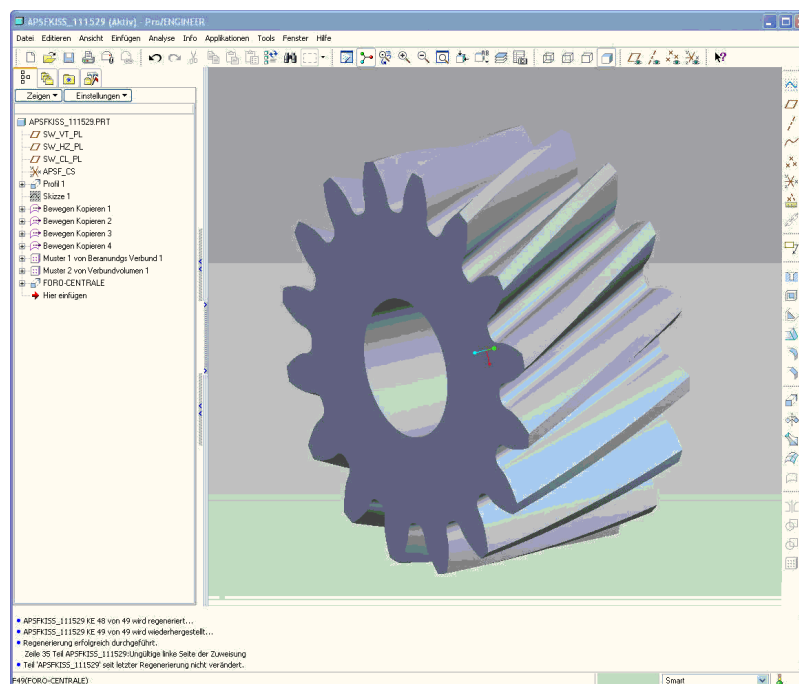


Рис. 3: Pro/ENGINEER

КОМПАС [19] компании Аскон. Отличительная черта данного продукта — ориентированность на поддержку стандартов чертежей и спецификаций, принятых в российской промышленности. Программа имеет удобный интерфейс, разнообразные способы и режимы построения графических примитивов, возможность управления порядком отрисовки графических объектов, средства создания параметрических моделей для часто применяемых типовых деталей, а также встроенную систему проектирования спецификаций.

2.3.4. SolidWorks

Нельзя не отметить известный программный комплекс SolidWorks [11]. В нём используется принцип трехмерного твердотельного и поверхностного параметрического проектирования, что позволяет конструктору создавать объемные детали, по которым создаются двумерные чертежи и спецификации в соответствии с определёнными требованиями. Особенно ценным преимуществом является возможность вносить изменения параметров прямо в трехмерную модель. Все элементы в режиме реального времени изменятся в соответствии с новыми данными,

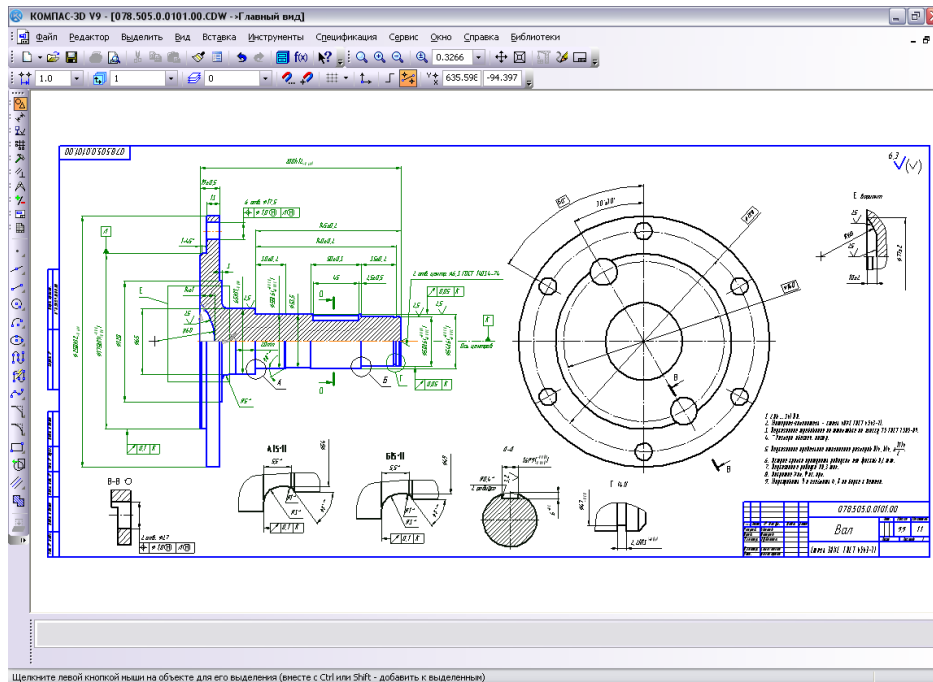


Рис. 4: КОМПАС

и все изменения тут же будут отражены в проектной документации.

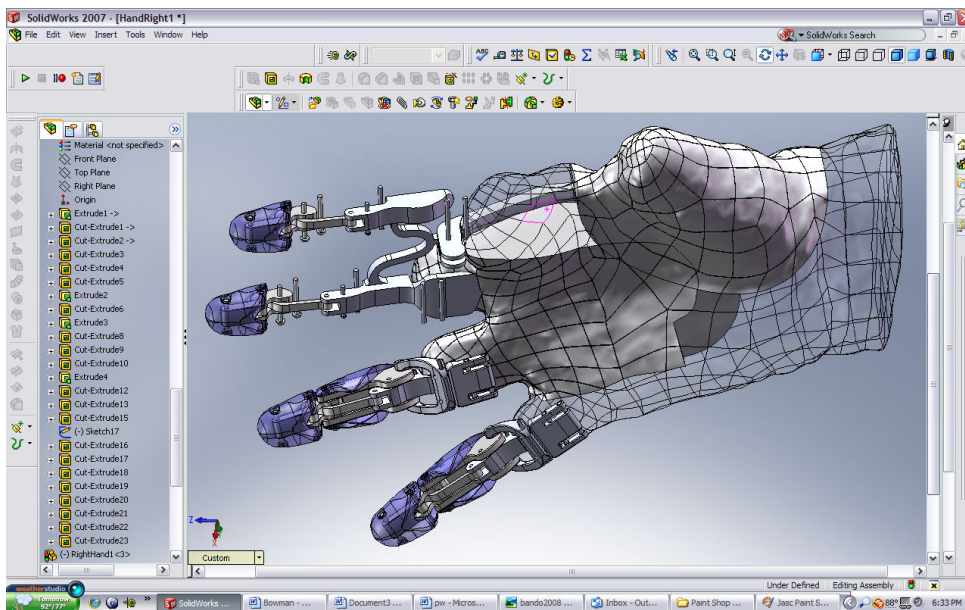


Рис. 5: SolidWorks

2.4. Актуальность задачи создания параметрического редактора для ПК ELCUT

Как говорилось ранее, в связи с разработкой новой версии программного комплекса ELCUT появилась необходимость в геометрическом редакторе. Так как ранее в редакторе ELCUT не было возможности параметризации объектов, было решено использовать параметрический редактор, а так как комплекс направлен на решение плоских задач, редактор должен быть двумерным. Стоит уделить внимание вопросу: почему не использовать готовое геометрическое ядро, зачем разрабатывать прототип самостоятельно? Тому существует несколько причин.

Во-первых, большинство существующих программных продуктов слишком дорогие. ELCUT отличается доступностью на рынке САЕ систем. Если использовать дорогостоящее геометрическое ядро, стоимость ELCUT значительно возрастёт, и часть пользователей может отказаться от данной системы. Целесообразно использовать недорогой редактор.

Во-вторых, ELCUT отличается простотой интерфейса, что объясняет популярность в учебных заведениях среди студентов и преподавателей. Большинство готовых ядер имеют достаточно сложный интерфейс. Использование таких ядер приведет к тому, что пользователю придется сначала разбираться в редакторе, что отнимет время от основной цели — решения необходимой задачи.

В-третьих, существующие ядра в большинстве своем трёхмерные, следовательно более сложные и "тяжёлые". Использование таких объёмных трёхмерных пакетов и библиотек для ELCUT, может снизить его производительность. Необходим двумерный геометрический редактор.

Таким образом, с учётом специфических потребностей программного комплекса ELCUT было принято решение разработать экспериментальный прототип геометрического редактора и попробовать реализовать один из возможных интерфейсов.

3. Проектирование системы

3.1. Выбор используемых технологий

Программа написана на C# в интегрированной среде разработки Microsoft Visual Studio [12]. Данный выбор обусловлен тем, что этот продукт позволяет создавать приложения с графическим интерфейсом наиболее удобно и быстро.

Для разработки данной программы также рассматривались другие варианты: NetBeans [13] и IntelliJ IDEA [6]. Реализация с использованием NetBeans осложнена тем, что для разработки программ и успешной инсталляции и работы самой среды должен быть предварительно установлен Sun JDK или J2EE SDK подходящей версии. Похожая ситуация и с IntelliJ IDEA, но здесь ещё появляются неудобства с импортом классов и с появлением в этой связи ошибок компиляции. Всё это требует дополнительного времени и предварительной установки программ для решения проблем при запуске приложения на другом ПК.

3.2. Проектирование функциональности

В рамках данной работы было проведено проектирование функциональности системы, в ходе которого необходимо было решить следующие задачи:

- отобразить геометрические примитивы, которые будут использоваться в редакторе;
- отобразить основные возможности редактирования;
- выбрать способы параметризации, которые будут реализованы в данной версии программы.

Ниже эти задачи рассмотрены более подробно.

3.2.1. Геометрические примитивы

Учитывая особенности программного комплекса ELCUT, описанные в разделе 2.4, было принято решение использовать следующие примитивы:

- точки;
- отрезки (две точки, соединённые линией);
- окружности.

Такого минимального набора элементов достаточно для того, чтобы описать основную часть двумерной геометрии, которая необходима при разработке данного редактора.

Ещё одним важным геометрическим элементом в ПК ELCUT является дуга окружности, но этот элемент не был включен в набор основных примитивов, так как данная версия интерфейса является экспериментальной. Поддержку дуг можно будет легко добавить в одной из следующих версий системы.

3.2.2. Понятие фигуры

Для получения более сложных объектов, было принято решение объединять описанные выше элементы в группы. Это можно будет сделать следующим образом:

- создать примитивы;
- составить из них необходимую конфигурацию;
- зафиксировать положение примитивов друг относительно друга.

Такие конфигурации примитивов в данном редакторе называются фигурами. Объединяя элементы в группы можно получать произвольные фигуры или основные геометрические фигуры (например, треугольники, прямоугольники, трапеции). Созданная фигура должна рассматриваться как самостоятельный объект, то есть иметь такие же воз-

возможности, как примитивы. При перемещении фигуры сохраняется относительное положение между примитивами, то есть, при изменении координаты одного из них, меняется положение всей группы.

3.2.3. Возможности редактирования

Было решено реализовать следующие возможности редактирования геометрических примитивов и фигур:

- добавление;
- удаление;
- перемещение.

Для фигур должна быть дополнительно реализована возможность их изменения (добавление новых элементов в фигуру или удаление существующих).

Важное значение в геометрических редакторах имеет копирование. Данная опция не реализована, так как было решено в данной версии обеспечить только базовые возможности.

3.2.4. Параметризация

В данной версии редактора было решено реализовать следующие виды параметров:

- угол между отрезком и одной из осей координат;
- размер вдоль заданного направления;
- размер по всем направлениям.

Должна быть возможность привязать параметр к определенной фигуре. Для этого нужно сначала создать параметр (задать название, тип и значение), затем выбрать пару точек из фигуры, тем самым задавая отрезок, для которого будет применен параметр. Эту процедуру можно будет проделывать для нескольких фигур, то есть один и тот же

параметр будет сразу применён к разным фигурам. Далее необходимо применить параметризацию, после чего фигура изменит свою геометрию в зависимости от параметра. Более подробно параметры описаны ниже.

- Угол между отрезком и одной из осей координат

Фигура повернётся вокруг начала заданного отрезка так, что угол между осью Ox и этим отрезком станет равным значению параметра, как показано на рисунке 6.

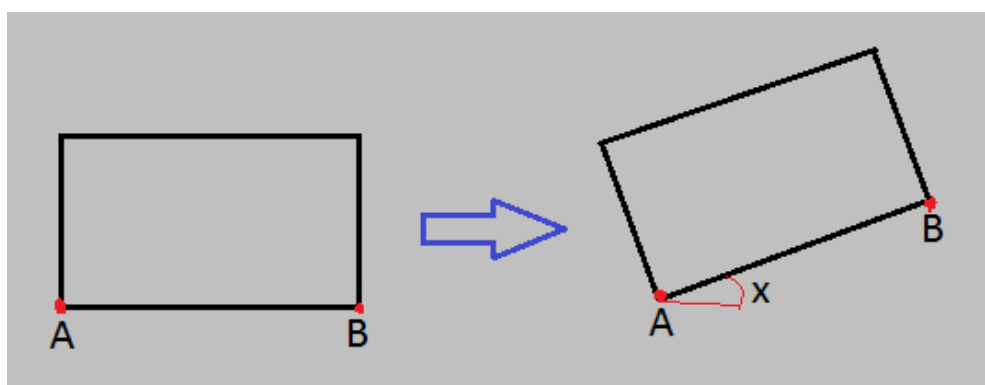


Рис. 6: Угол между отрезком и осью ox

- Размер вдоль заданного направления

Фигура растянется вдоль оси, образуемой заданным отрезком (начальная точка остается на месте), до фиксированного значения, как показано на рисунке 7.

- Размер по всем направлениям

Строится подобная фигура, с коэффициентом подобия равным отношению значения параметра к длине заданного отрезка, как показано на рисунке 8 (начальная точка остается на месте).

- Параметризация окружностей

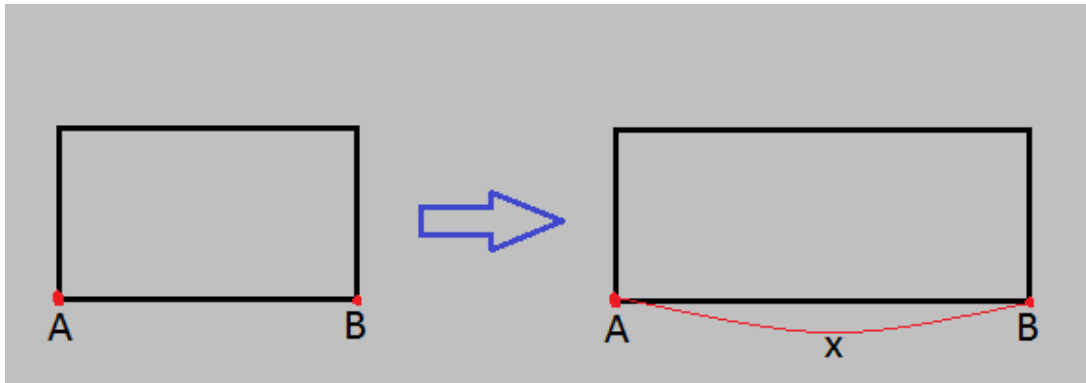


Рис. 7: Размер вдоль отрезка АВ

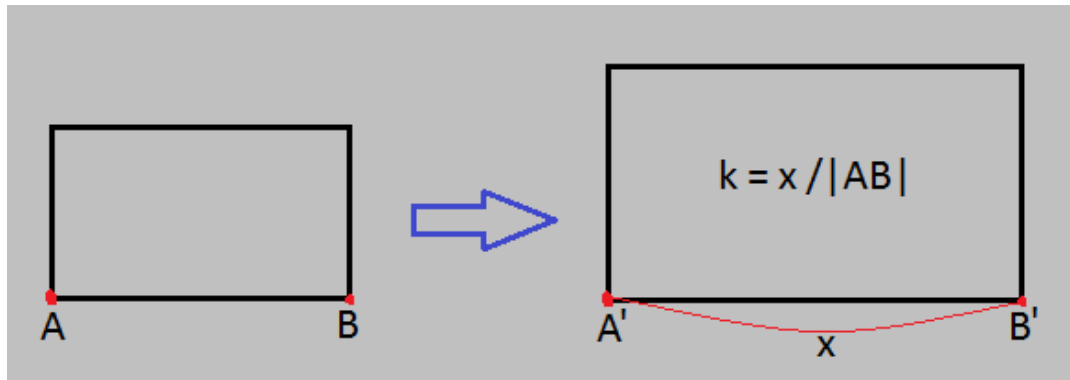


Рис. 8: Размер по всем направлениям с коэффициентом k

Существует специфический случай — применение параметров для фигуры, в составе которой есть окружность. Окружности могут быть с фиксированным радиусом и с масштабируемым радиусом. Если параметризация применяется к окружности первого типа, то радиус не меняется, а изменяются координаты центра окружности. Для окружностей второго типа параметризация применяется и к координатам центра, и к радиусу.

- Операции с параметрами

Естественно, значение параметра можно изменять, после чего новое значение должно быть применимо ко всем привязанным фигурам. Также должна быть возможность удалить параметр или удалить из параметра привязанную пару точек фигуры, чтобы параметр и фигура перестали быть связанными.

3.2.5. Проверка совместимости

Во избежание ошибок построения было решено использовать проверку совместимости параметров — некоторые ограничения на применение параметров. Установлены следующие правила:

- Для одной фигуры одновременно применимы растяжения вдоль заданной оси только в том случае, если между ними угол 90° .
- Для одной фигуры применим только один параметр масштабирования.
- Для одной фигуры применим только один параметр фиксации угла.
- Для одной фигуры могут быть одновременно применимы растяжение и поворот, либо масштабирование и поворот.

3.2.6. Преобразования

Помимо параметров было решено использовать операции, которые называются "преобразования". Они чем-то похожи на параметры, но имеют другое назначение и конечную цель. Сначала создается преобразование (название, значение и тип), затем фиксируется пара точек из фигуры, для которой необходимо выполнить выбранное преобразование. Были введены следующие преобразования:

- Поворот на заданный угол по часовой и против часовой стрелки: фигура поворачивается на заданный угол.
- Увеличение на заданную величину: строится подобная фигура с коэффициентом подобия равным отношению длины отрезка к сумме значения параметра и длины отрезка.
- Растяжение вдоль заданной оси на заданную величину: фигура растягивается вдоль заданной отрезком оси на величину N .

В отличие от параметров преобразования не фиксируют конечное положение фигуры. При повторном выполнении преобразования фигура снова меняет свою геометрию. Эта возможность не имеет отношения к параметризации, но практика показала, что такие дополнительные возможности редактирования объектов очень уместны и часто используются.

3.3. Проектирование пользовательского интерфейса

При разработке интерфейса необходимо было решить несколько вопросов:

1. Как отображать информацию о примитивах и фигурах?

Фигура должна отличаться от примитива цветом, а информация отображаться при наведении курсора на элемент при помощи всплывающего окна.

2. Как отображать информацию о параметрах?

Необходимо выводить список параметров, в котором можно выбрать один из них, при этом нужно выделить фигуры, связанные с этим параметром.

3. Как на рисунке показать ограничения, связанные с параметром?

В данном случае целесообразно показывать отрезок, связанный с определённым параметром. Различать параметры можно с помощью цветов отрезков.

4. Реализация

4.1. Описание пользовательского интерфейса

Программа позволяет в интерактивном режиме создавать, удалять и перемещать геометрические примитивы (точки, отрезки и окружности). Есть возможность создавать фигуры, состоящие из набора примитивов. Их можно параметризовать, после чего изменится геометрия фигуры в зависимости от параметра. Рассмотрим подробнее, как всего этого добиться с помощью данной программы.

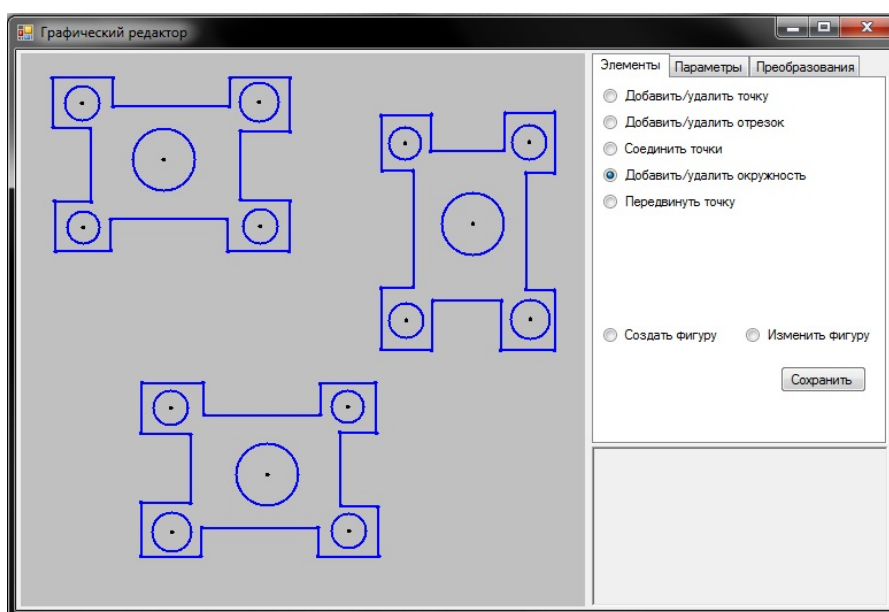


Рис. 9: Пример работы программы: основное окно.

На рисунке 9 представлено главное окно программы. Слева расположена специальная область, в которой можно создавать примитивы, фигуры и задавать параметризацию с помощью мыши. Справа располагается меню с несколькими вкладками и окно с подсказками для пользователя.

4.1.1. Вкладка "Элементы"

Для начала справа необходимо выбрать один из режимов создания примитивов во вкладке "Элементы":

1. Добавить/удалить точку

В этом режиме при нажатии левой кнопки мыши появляется точка, при нажатии правой кнопки по существующей точке, она удаляется. При наведении курсора на точку, появляется окошко с информацией: номер и координаты точки.

2. Добавить/удалить отрезок

Для того, чтобы добавить отрезок, надо нажать левую кнопку, не отпуская вести мышь до нужного места и отпустить кнопку. Для удаления существующего отрезка необходимо навести на него курсор и нажать правую кнопку мыши. При наведении курсора на отрезок, появляется окошко с информацией: номер начала и номер конца отрезка.

3. Соединить точки

Этот режим используется для того, чтобы добавить отрезок между двумя точками. Пользователь должен навести мышь на существующую точку, нажать левую кнопку, затем проделать то же самое со второй точкой, и появится линия, соединяющая эти точки.

4. Добавить/удалить окружность

Для того, чтобы добавить окружность, нужно нажать левую кнопку, не отпуская вести мышь до нужного места и отпустить кнопку. Для удаления существующей окружности необходимо навести на неё курсор (чтобы он попадал в пределы окружности) и нажать правую кнопку мыши. При наведении курсора на окружность, появляется окошко с информацией: номер, координаты центра и радиус окружности.

5. Передвинуть точку

Этот режим используется для изменения положения точки и всех примыкающих к ней отрезков, перемещения окружности (если точка — это центр окружности) или перемещения фигуры (если выбранная точка входит в состав фигуры). Для этого необходимо навести курсор на существующую точку, зажать левую кнопку мыши и перемещать

точку на новое место, затем отпустить кнопку.

6. Создать фигуру

Данный режим позволяет создавать совокупность объектов, фиксируя относительное положение этих объектов. Для этого нужно наводить курсор и нажимать левую кнопку мыши поочередно на те точки, которые нужно связать в единый объект, после чего нажать кнопку «Сохранить». Точки, которые входят в состав какой-либо фигуры становятся красного цвета. Каждый примитив может входить в состав только одной фигуры. Теперь созданную фигуру можно перемещать целиком, перейдя в режим «Передвинуть точку». При наведении курсора на какой-либо примитив, который входит в состав фигуры, отображается дополнительная информация: номер фигуры.

7. Изменить фигуру

Существующую фигуру можно редактировать следующим образом: сначала необходимо выбрать фигуру (навести курсор на любую из точек, которые принадлежат данной фигуре и нажать левую кнопку мыши), затем по нажатии левой кнопки мыши добавляются существующие точки, а по нажатии правой кнопки точки удаляются из фигуры. Для сохранения фигуры необходимо нажать кнопку «Сохранить».

4.1.2. Вкладка «Параметры»

Далее, перейдя во вкладку «Параметры» (рисунок 10), можно совершать следующие манипуляции с параметрами:

1. Создание нового параметра.

Необходимо нажать кнопку «Добавить» и ввести название, значение и выбрать тип параметра в появившемся диалоговом окне. Как говорилось в разделе 3.2.4, существует три типа параметров:

- угол между отрезком и одной из осей координат;
- размер вдоль заданного направления;

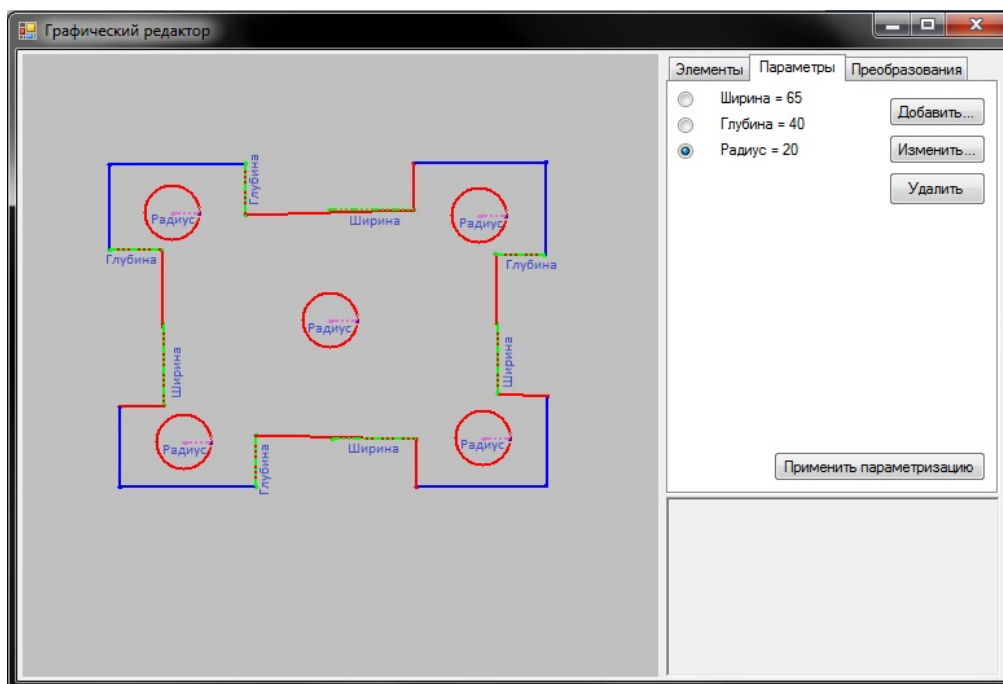


Рис. 10: Пример работы программы: добавление параметров.

- размер по всем направлениям.

Для угла значение задается в градусах, для остальных параметров — в пикселях. Вновь созданные параметры будут появляться в текущей вкладке в списке, как показано на рисунке 10.

2. Изменение выбранного параметра.

Уже существующие параметры можно изменять. Для этого нужно выбрать параметр в списке (с помощью `RadioButton`, как на рисунке 11), нажать кнопку «Изменить» и в появившемся диалоговом окне ввести новое название и значение параметра (тип существующего параметра изменять нельзя). Новые значения параметра после сохранения автоматически применятся ко всем фигурам, с которыми связан этот параметр.

3. Удаление выбранного параметра.

Необходимо выбрать параметр в списке и нажать кнопку «Удалить», после чего появится диалоговое окно для подтверждения действия. При удалении параметра все фигуры, связанные с ним, перестают быть па-

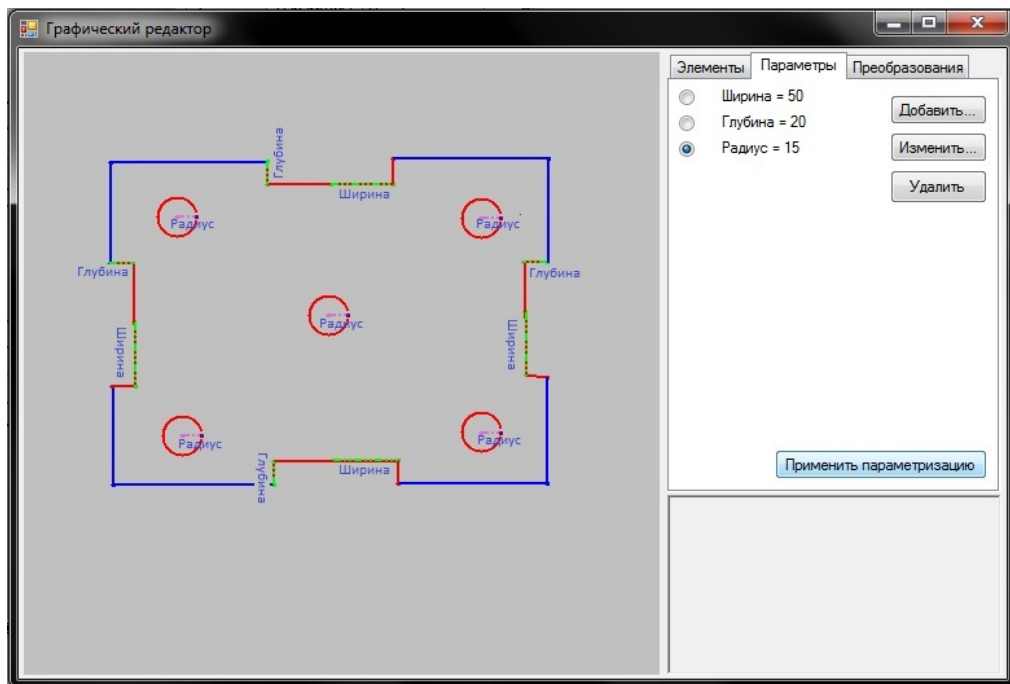


Рис. 11: Пример работы программы: изменение параметров.

раметризованными.

4. Параметризация фигуры

После создания параметров нужно связать их с определенными парами точек, к которым этот параметр в дальнейшем будет применён. Параметризация может применяться только к фигурам (к отдельным примитивам не может). Для этого нужно выбрать параметр в списке и двойным щелчком левой кнопки мыши выбрать сначала одну точку, затем вторую, после чего точки поменяют цвет в зависимости от типа параметра. Один параметр можно применять ко многим фигурам или даже несколько раз к одной и той же фигуре (если позволяет описанная в разделе 3.2.5 проверка совместимости). Для выбранного параметра можно как добавлять фигуру, так и удалять уже привязанную. Для этого нужно дважды щелкнуть правой кнопкой мыши по любой точке из пары. По нажатии кнопки «Применить параметризацию» все фигуры, связанные с каким-либо параметром изменят свою геометрию в соответствии со значением и типом соответствующего параметра.

4.1.3. Вкладка "Преобразования"

Как говорилось ранее, над фигурами можно совершать "преобразования", перейдя во вкладку с соответствующим названием. Преобразования аналогичны параметрам в смысле изменения геометрии фигуры, но отличаются тем, что их можно применять неоднократно, и после каждого применения геометрия меняется (не фиксируется, как в случае параметров).

В целом, над преобразованиями можно совершать такие же манипуляции, как и с параметрами:

1. Создание нового преобразования.
2. Изменение выбранного преобразования.
3. Удаление выбранного преобразования.
4. Изменение фигуры с помощью преобразования.

4.2. Описание реализации

Данная программа состоит из главного класса Form1 [17] — окна приложения, в котором хранятся списки созданных объектов. На главном окне формы располагается элемент управления pictureBox1 (экземпляр класса PictureBox [17]) — специальная область, в которой отображаются геометрические объекты и создаются с помощью мыши. Каждый примитив является экземпляром класса Point (точка), Line (линия) или Circle (круг). Конфигурации примитивов являются экземплярами класса Figure (фигура).

У основного класса есть методы — обработчики событий [17]:

- MouseDown — нажатие клавиши мыши;
- MouseUp — отпускание клавиши мыши;
- MouseMove — передвижение курсора;
- MouseDoubleClick — двойной щелчок кнопки мыши;
- MouseClick — щелчок кнопки мыши;

- Paint – перерисовка области, в которой создаются элементы (PictureBox);
- нажатие кнопок, расположенных на форме.

Кроме этого, в классе есть вспомогательные функции (например, `is_in_list`), в которые для удобства вынесены часто выполняемые операции (например, проверка, существует ли точка в списке с текущими координатами) и различные вспомогательные классы для обеспечения удобной структуры хранения данных, а именно:

- Point
- Line
- Circle
- Figure
- Parameter

Особого внимания заслуживают функции, осуществляющие преобразования координат точек для обеспечения параметризации фигур. Для всех фигур параметризация выполняется следующим образом [25]:

1. Фиксируется отрезок, входящий в состав фигуры (точка А — начало, точка В — конец).
2. Выполняется сдвиг фигуры так, чтобы точка А оказалась в точке (0,0).
3. Выполняется поворот фигуры так, чтобы отрезок АВ стал параллельным оси Ох.
4. Выполняется необходимое преобразование фигуры (см. раздел 3.2.4).
5. Выполняется обратное пункту 3 преобразование.
6. Выполняется обратное пункту 2 преобразование.

4.3. Анализ результатов тестирования и опытной эксплуатации

Данная система была предложена для тестирования разработчикам ELCUT, сотрудникам ООО "ТОР". В результате тестирования были сделаны следующие выводы:

- Предложенный набор примитивов является достаточно удобным и может быть использован для простых случаев параметризации. В то же время необходимо развитие системы для более сложных элементов и способов параметризации.
- Интерфейс выделения параметров с помощью отрезков был признан удобным, но было высказано предложение добавить возможность временно "отключать" параметры и усовершенствовать систему отображения параметров (подсвечивать фигуры только для выбранного параметра).
- Введенные преобразования оказались очень удобными и полезными, так как с их помощью можно проводить предварительное исследование изменения геометрии фигур, перед тем, как вводить параметризацию.
- Были высказаны предложения по добавлению возможности фиксирования угла с произвольными направлениями, а не только с осями координат.
- Многие пришли к мнению, что для данного интерфейса было бы уместным использование "горячих клавиш". Например, при создании фигуры: мышкой выбирать набор примитивов, а сохранять фигуру по нажатию кнопки "Enter".

4.4. Основные результаты

Перечислим возможности, отличающие данную систему от других:

- пользовательский интерфейс;

- способы визуализации связи объекта и параметра;
- небольшой, но достаточно мощный набор средств параметризации;
- возможность изменения объектов при помощи "преобразований";
- проверка совместимости параметров.

Заключение

В рамках данной работы получены следующие результаты:

1. Разработан и реализован графический редактор с возможностью параметризации, с помощью которого можно создавать простые геометрические объекты, изменять их и удалять.

2. Проанализирован интерфейс программы. Сделан вывод, что для данного набора элементов и параметров он удобен, но требуются дальнейшие исследования для обобщения рассматриваемого подхода на более сложные виды параметрических связей (например, касательная к окружности).

3. Программа протестирована, на основе чего намечены пути исправления и модернизации функциональности приложения.

Список литературы

- [1] C3D Solver. — URL: <http://c3dlabs.com/ru/products/solver> (дата обращения: 16.03.2016).
- [2] Computer-aided design. — URL: https://en.wikipedia.org/wiki/Computer-aided_design (дата обращения: 07.01.2016).
- [3] Computer-aided engineering. — URL: https://en.wikipedia.org/wiki/Computer-aided_engineering (дата обращения: 07.01.2016).
- [4] ELCUT. — URL: <http://elcut.ru> (дата обращения: 29.11.2015).
- [5] FreeCAD: An open source parametric 3D CAD modeler. — URL: <http://www.freecadweb.org> (дата обращения: 08.04.2016).
- [6] IntelliJ IDEA the Java IDE. — URL: <https://www.jetbrains.com/idea> (дата обращения: 19.11.2015).
- [7] Open CASCADE. — URL: <http://www.opencascade.com> (дата обращения: 19.02.2016).
- [8] Parametric modeling. — URL: https://en.wikipedia.org/wiki/Solid_modeling#Parametric_modeling (дата обращения: 08.04.2016).
- [9] Parasolid. — URL: <https://ru.wikipedia.org/wiki/Parasolid> (дата обращения: 19.02.2016).
- [10] Pro/ENGINEER. — URL: <http://ru.ptc.com/product/creo/proengineer> (дата обращения: 08.04.2016).
- [11] SolidWorks. — URL: <http://www.solidworks.ru> (дата обращения: 08.04.2016).
- [12] Visual Studio — Microsoft Developer Tools. — URL: <https://www.visualstudio.com> (дата обращения: 19.11.2015).

- [13] Welcome to NetBeans. — URL: <https://netbeans.org> (дата обращения: 19.11.2015).
- [14] Автоматизированное конструирование. — URL: <http://plmpedia.ru/wiki/CAE> (дата обращения: 07.01.2016).
- [15] Будущее параметрического CAD'а. — URL: <https://habrahabr.ru/company/cloudinvent/blog/245835> (дата обращения: 07.01.2016).
- [16] Вариационное проектирование. — URL: http://plmpedia.ru/wiki/Вариационное_проектирование (дата обращения: 16.03.2016).
- [17] Документация по Visual Studio на MSDN. — URL: <https://msdn.microsoft.com/library/dd831853.aspx> (дата обращения: 09.11.2015).
- [18] Дятлов М. Н., Юдкин И. Ю., Шляховский А. А. Тенденции развития современных интегрированных систем проектирования. — Молодой ученый, 2015. — URL: <http://www.moluch.ru/archive/82/14848>.
- [19] Компас — 3D V16. Инструмент создателя. — URL: <http://kompas.ru> (дата обращения: 08.04.2016).
- [20] Метод конечных элементов. — URL: https://ru.wikipedia.org/wiki/Метод_конечных_элементов (дата обращения: 29.11.2015).
- [21] Параметризация. — URL: https://ru.wikipedia.org/wiki/Параметрическое_моделирование (дата обращения: 08.01.2016).
- [22] Параметрический подход к проектированию. — URL: <http://www.tflex.ru/about/publications/detail/index.php?ID=1177> (дата обращения: 07.01.2016).
- [23] Программа для архитектурного проектирования и черчения AutoCAD. — URL: <http://www.autodesk.ru/products/autocad-architecture/overview>.

- [24] Проектирование на базе предыстории. — URL: http://plmpedia.ru/wiki/Проектирование_на_базе_предыстории (дата обращения: 16.03.2016).
- [25] Роджерс Д., Адамс Дж. Математические основы машинной графики. — Мир, 2001. — С. 604.
- [26] Состав пакета ELCUT. — URL: http://elcut.ru/components_r.htm (дата обращения: 16.03.2016).
- [27] Трехмерное параметрическое моделирование на персональном компьютере. — URL: http://www.e-ng.ru/programmirovaniye_i_komp-ry/trekhmernoe_parametricheskoe.html (дата обращения: 27.12.2015).
- [28] Ядро геометрического моделирования. — URL: http://plmpedia.ru/wiki/Ядро_геометрического_моделирования (дата обращения: 19.02.2016).