

СЕКЦИЯ 1

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

Среда, 23 апреля 2014 г., читальный зал преподавателей (ауд.229), гл. корпус МГТУ им.Н.Э.Баумана.

Начало в 10.00.

Председатель: профессор, д.ф-м.н. Карпенко А.П.

Руководитель экспертной комиссии: к.т.н., доцент Федорук В.Г.

Ученый секретарь: к.т.н., доцент Власов А.И.

Экспертная комиссия:

А.Н.БОЖКО, Т.М.ВОЛОСАТОВА, Д.М.ЖУК, В.Б.МАНИЧЕВ, В.А.МАРТЫНЮК, В.А.ТРУДОНОШИН, С.В.ГРОШЕВ, С.Ю.КНЯЗЕВА, С.В.РОДИОНОВ, Е.В.ФЕДОРУК, С.А.ШЕСТАКОВ, В.В.БЕЛОУС, ПИВОВАРОВА Н.В.

В рамках работы секции будет осуществлен первый этап конкурсного отбора проектов для участия в конкурсе инновационных проектов МГТУ им.Н.Э.Баумана.

Критериями конкурсного отбора участников молодежного научно-инновационного конкурса являются:

1. Уровень инновационности идеи (предложения, метода, способа ...).

1.1. Идея должна быть новой, впервые сформулированной именно самим номинантом. В этом смысле все номинанты равны.

1.2. Уровень наукоемкости тем выше, чем более:

- основательны научные исследования, в результате которых она появилась;

- основательны дальнейшие научные исследования, необходимые для ее реализации.

1.3. Техническая значимость тем выше, чем большее влияние ее реализация окажет на уровень техники. «Пионерные» идеи (изобретения) – пенициллин, лазер, синтез алмазов - открывают новые отрасли науки и техники. Идеальная по инновационности идея неожиданна для рынка. Поэтому она им не может быть сейчас востребована, она сама формирует новую потребность и нишу рынка.

Высокий технический уровень имеют решения, например, многоотраслевого использования.

Оригинальные технические решения дают новые принципы решения известной задачи.

Есть решения, позволяющие решить проблему еще одним, дополнительно к известным, способом. Но и в этом случае, если уже известно 10 способов решения задачи, то новый способ может и не давать существенных преимуществ в решении задачи или давать их только в очень ограниченном по масштабам применения числе случаев.

1.4. Масштабность использования предложения тоже может сильно различаться – от решения локальной задачи одного местного потребителя до ...

1.5. Срок превращения идеи в конечный продукт с выходом его на рынок: новизна, рискованность идеи, объем необходимых научных исследований не позволяют уложиться в 2-3 года, но и не требуют 10-15 лет

1.6. Идея тем актуальней, чем меньше вероятность того, что за 5-7 лет в результате научных исследований появятся и «раскрутятся» до продукта другие более эффективные пути решения задачи.

Победитель сам организует работу по привлечению необходимого финансирования. Инновационная программа МГТУ им.Н.Э.Баумана оказывает ему поддержку на начальном этапе работы над его идеей.

ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ МАЛОВЫСОТНОГО ПОЛЕТА ВЕРТОЛЕТА ПО ТЕКУЩИМ ДАННЫМ БОРТОВЫХ НАВИГАЦИОННЫХ СИСТЕМ

Андросов Г.В.

Научный руководитель: д.т.н. Бабиченко А.В.

ОАО «РПКБ», Раменское, Россия

INFORMATION ENSURING FOR HELICOPTER LOW-ALTITUDE FLIGHT WITH BOARD NAVIGATION SYSTEMS DATA

Androsoy G.V.

Supervisor: Dr., Babichenko A. V.

RDC, Ramenskoye, Russia

Аннотация

В статье рассматриваются методы информационного обеспечения режима маловысотного полета вертолета по текущим данным систем обзора пространства. Подробно исследовано моделирование облета рельефа в вертикальной плоскости. Кратко представлен алгоритм обеспечения режима маловысотного полета. В заключении представлены результаты моделирования и их анализ.

Abstract

This article discusses methods of information ensuring for mode of low-altitude helicopter flight on current data by space reviewing systems. Modeling vertical plane's relief overflight is studied in detail. Briefly present an algorithm of ensuring a low-altitude flight mode. In conclusion, the results of simulation and analysis are presented.

Введение

Задача маловысотного полета (МВП) вертолета возникает как следствие из задачи полета вертолета по заданной траектории. В настоящее время на ряде российских и зарубежных вертолетов реализован режим полета по заданной траектории. Реализация данного режима подразумевает наличие цифровой карты местности (ЦКМ) и заданной на ней траектории полета. В контексте данного подхода, режим МВП представляет собой полет по траектории, проложенной на некоторой заданной высоте над рельефом. Однако, данный подход встречает ряд сложностей, в числе которых возможное отсутствие или устаревание цифровой карты данной местности и необходимость оперативного изменения траектории полета при сохранении режима облета рельефа в вертикальной плоскости.

Подробное исследование динамики полета вертолета и его САУ приведено в [1], общие вопросы автоматизации МВП – в [2], при этом построение оптимальной траектории МВП при наличии ЦКМ – в [3]. В данной работе подробнее рассматривается обеспечение режима МВП в отсутствии ЦКМ, только по данным бортовых систем обзора пространства.

Целью работы явилось разработка метода обработки информации от бортовых обзорных систем и формирование входных данных на пилотажно-навигационный комплекс (ПНК) вертолета. В ходе работы были созданы модели «Вертолет+ПНК», «Рельеф», «Данные средств обзора» в среде MATLAB Simulink. С помощью них был смоделирован режим облета рельефа в вертикальной плоскости.

Задача данной работы поставлена следующим образом:

1. Определить состав средств обзора пространства и данных, получаемых от них;
2. Определить состав данных, поступающих на ПНК вертолета и алгоритм их расчета;
3. Разработать модели рельефа местности, средств обзора и вертолета с пилотажно-навигационным комплексом;
4. Разработать модель режима МВП вертолета и проанализировать результаты.

1 Состав средств обзора

Задача облета рельефа в вертикальной плоскости состоит в выдерживании заданной высоты H_z следования над рельефом местности. Для определения относительной высоты над рельефом местности $H_{отн}$ необходимо использовать высотомер. Положение луча высотомера в горизонтированной системе координат принимается неизменным и совпадающим с нормалью к горизонту. Однако, при наличии препятствий на пути следования вертолета, одного высотомера оказывается недостаточно. Для определения наличия препятствий на пути следования и расстояния до них на вертолете должен быть установлен дальномер. Положение луча дальномера относительно горизонтированной системы координат принимается постоянным и равным $\psi=20^\circ$. С дальномера снимаются показания D – дальности по лучу до поверхности рельефа. Угол тангажа вертолета принимается постоянным. Горизонтальная скорость вертолета V_x в режиме МВП также принимается постоянной.

2 Состав данных, поступающих на ПНК вертолета

На ПНК вертолета поступают данные о рассогласовании по относительной высоте полета:

$$\Delta H_{МВП} = H_{отн} - H_z$$

Для облета рельефа в вертикальной плоскости необходимо управлять вертикальной скоростью. Условием облета рельефа является выдерживание вертолетом некоторого угла наклона траектории θ . Этот угол должен совпадать с углом наклона рельефа δ . Стоит задача измерения угла наклона рельефа.

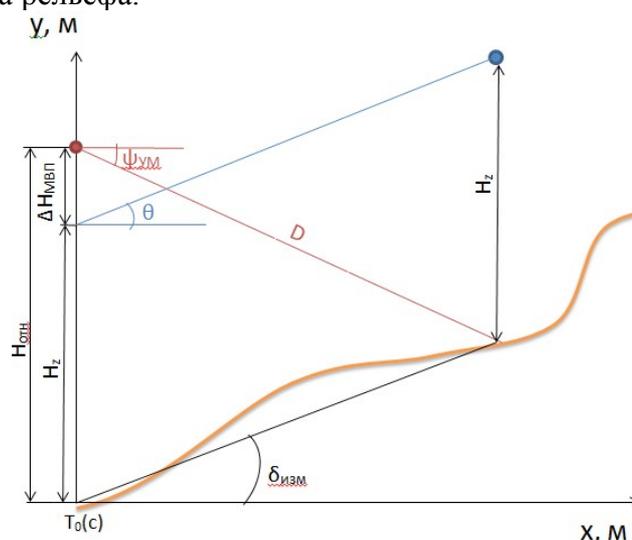


Рисунок 1 - Схема измерений

Угол наклона рельефа измеряется в между прямой, соединяющей точки пересечения луча высотомера и дальномера с поверхностью рельефа, и линией горизонта.

Таким образом:

$$\delta_{изм} = \arctg \frac{\Delta y}{\Delta x} = \arctg \frac{H_{отн} - D \cdot \cos \psi}{D \cdot \sin \psi}$$

$\delta_{изм}$ – измеренный угол наклона рельефа. Условие облета рельефа: $\theta = \delta_{изм}$.

Требуемая вертикальная скорость полета:

$$V_{yz} = V_x \cdot \operatorname{tg} \theta = V_x \cdot \operatorname{tg} \delta_{изм} = V_x \cdot \frac{H_{отн} - D \cdot \cos \psi}{D \cdot \sin \psi}$$

Рассогласование по вертикальной скорости:

$$\Delta V_y = V_y - V_{yz}$$

Таким образом, необходимо разработать модель «Вертолет+ПНК», управляемую по рассогласованию по относительной высоте и по вертикальной скорости.

3 Модель режима МВП

Модель МВП вертолета создана в среде MATLAB Simulink и состоит из трех основных элементов:

- модель «Рельеф»;
- модель «Данные от средств обзора».
- Модель «Вертолет+ПНК».

$$\dot{H} = 31.28\phi^2 + 156.4\phi^3 - 7.6 \cdot 10^{-3}\dot{H}^2 - 9.8,$$

- где ϕ – общий шаг несущего винта.

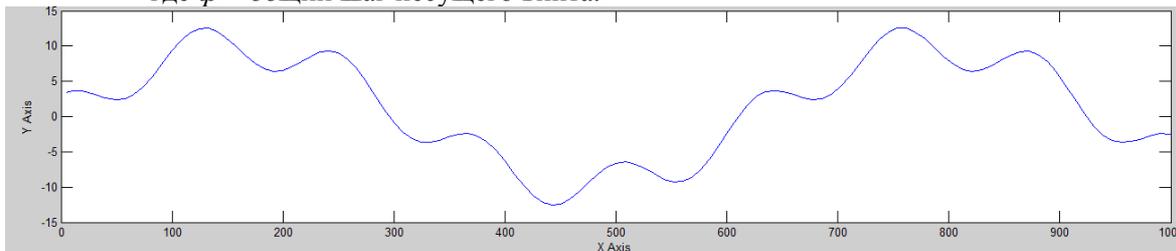


Рисунок 2 - Модель рельефа местности

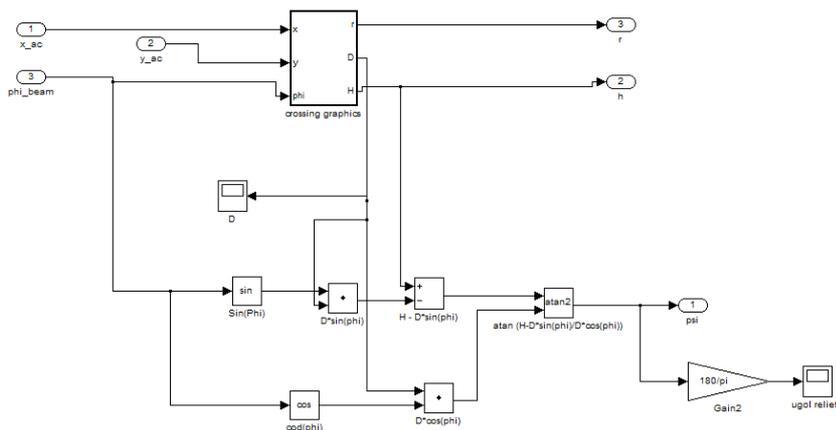


Рисунок 3 - Модель получения данных от высотомера и дальномера

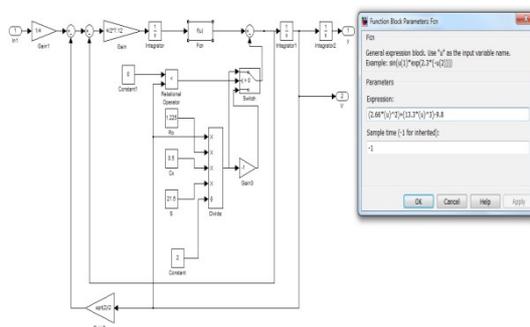


Рисунок 4 - Модель вертолета с ПНК

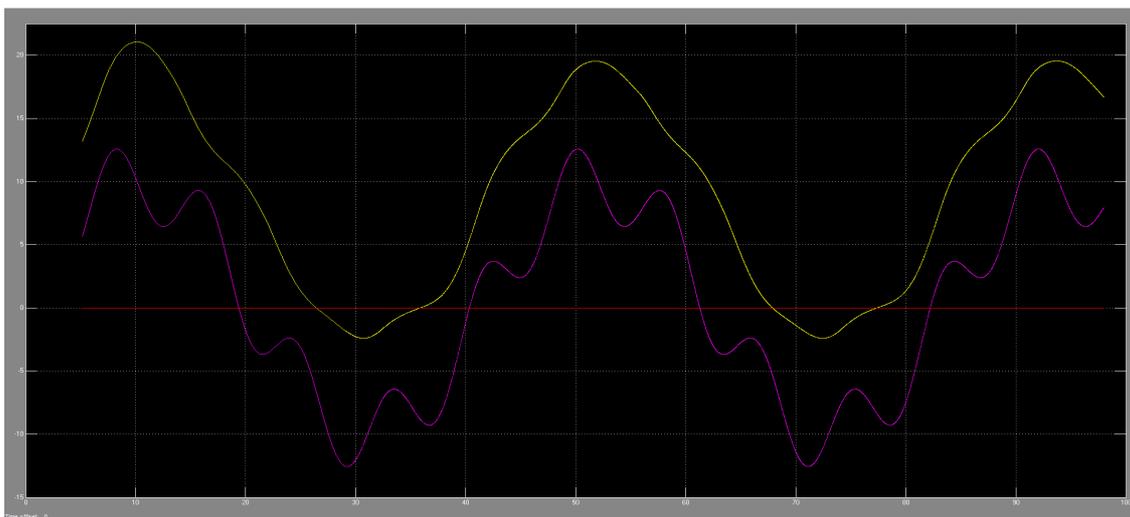


Рисунок 5 - Модель режима МВП вертолета. Пурпурный график – рельеф, желтый график – траектория вертолета

Коэффициенты рассчитаны, исходя из массо-габаритных и динамических характеристик реального вертолета с пилотажно-навигационным комплексом.

Заключение

Математическое моделирование режима МВП вертолета подтверждает работоспособность разработанного алгоритма его обеспечения. Как видно из графиков на рисунке 5, вертолет огибает на заданной высоте $H_z = 8$ м с максимальным сближением в 5 метров.

Преимущество использования данного алгоритма заключается в его простоте и независимости от поперечного движения. Данный алгоритм избавляет от необходимости построения траектории полета, что при фиксированном луче дальномера занимает достаточно длительное (6-10 секунд в зависимости от типа рельефа) время и пользуется только текущими данными, без необходимости их накопления, что позволяет оперативно управлять курсом вертолета.

Недостатком данного алгоритма является низкая точность измеряемого угла наклона рельефа, а следовательно и низкая точность выдерживаемого угла наклона траектории. Таким образом, рельеф, попадающий между лучами высотомера и дальномера, «сглаживается», что негативно сказывается на точности управления. Прочие следствия данного «сглаживания» будут подробнее исследованы в дальнейших работах.

Литература

1. Колоколов С.Н., Коновалов А.П., Куратов В.А. Динамика управляемого движения вертолета. - М.: Машиностроение. 1987г. 144 с
2. Никифорова Л.Н., Яковлев К.С. Маловысотный полет вертолета и проблемы его автоматизации. М.: Искусственный интеллект и принятие решений, 3/2009. – С. 42-28.
3. Никифорова Л.Н. Оптимальное управление в построении траекторий перелета вертолета в заданную точку пространства. М.: Программные средства: теория и приложения, №2(11), 2012 г., С. 61-75.

ПРОЕКТИРОВАНИЕ РЕЗОНАТОРОВ И ТАКТОВЫХ ГЕНЕРАТОРОВ В САПР COVENTOR MEMS+

Баранов И.Д., Тюнькин Д.К.

Научные руководители: к.т.н., доцент Жук Д.М., к.т.н., доцент Маничев В.Б.

МГТУ им. Н.Э.Баумана, кафедра РКБ, Москва, Россия

DESIGN OF RESONATORS AND CLOCK GENERATORS IN CAD COVENTOR MEMS+

Baranov I.D., Tunkin D.K.

Supervisors: Ph.D. Zhuk D.M., Ph.D. Manichev V.B.

BMSTU, Moscow, Russia

Аннотация

В статье рассматриваются вопросы автоматизированного процесса проектирования МЭМС резонаторов и тактовых МЭМС генераторов. Проведен сравнительный анализ кварцевых и МЭМС резонаторов. Выполнен обзор средств автоматизированного проектирования системы Coventor MEMS+.

Abstract

The article considers the questions of the automated design process MEMS resonator and clock MEMS generators. A comparative analysis of quartz and MEMS resonator. There is a review of computer-aided design system Coventor MEMS+.

Введение

В настоящее время тактовые генераторы находят широчайшее применение в современной технике. Они используются в технике радиосвязи, радиолокации измерительной и вычислительной технике, телеметрии, в бытовых приборах таких, как часы, магнитофоны, телевизоры и др. Их использование позволяет создать надежную радиоаппаратуру высокой точности, малых габаритов, простую в эксплуатации и с малой потребляемой мощностью.

1 Тактовые кварцевые и МЭМС генераторы

Тактовый кварцевый генератор – автогенератор с колебательной системой, в состав которой входит кварцевый резонатор, который представляют собой особым образом вырезанную и отшлифованную пластину натурального или искусственного кварца. Кварц - пьезоэлектрик, поэтому упругие колебания кристалла могут быть вызваны приложением электрического поля, а эти колебания, в свою очередь, генерируют напряжение на гранях кристалла. В этом случае кристалл ведет себя как RLC-элемент, эквивалентная схема которого приведена на рисунке 1.

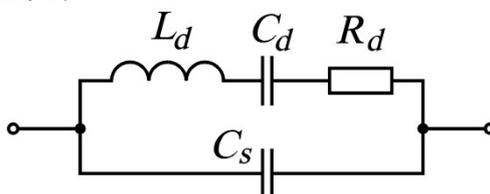


Рисунок 1 – Эквивалентная схема кварцевого резонатора

Два конденсатора эквивалентной схемы дают пару близко расположенных резонансных частот – последовательного и параллельного контура, отличающихся друг от друга не более чем на 1%. В целом кварцевый резонатор ведет себя как резонансный контур с высокой добротностью и высокой стабильностью параметров. При включении резонатора в положительную обратную связь и выполнении условия баланса амплитуд на резонансной частоте возникают автоколебания.

Основными схемами включения кварцевого резонатора являются генератор Пирса и кварцевый генератор Колпитца. Первый представляет собой генератор синусоидальных колебаний на полевом транзисторе. За счет кварцевого резонатора фаза выходного сигнала изменяется на 180° , т.е. суммарный сдвиг фазы по отношению к сигналу на затворе достигает 2π , что приводит к возникновению колебаний на резонансной частоте кварца. Другая схема (рис.3) представляет собой аналог генератора Колпитца, в котором LC – контур заменен кварцевым резонатором.

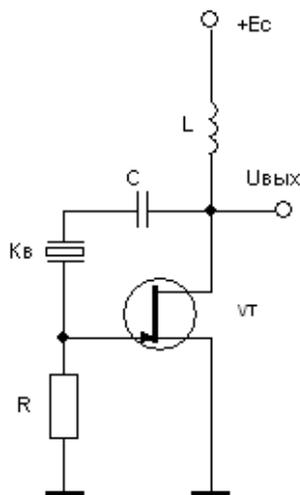


Рисунок 2 – Генератор Пирса

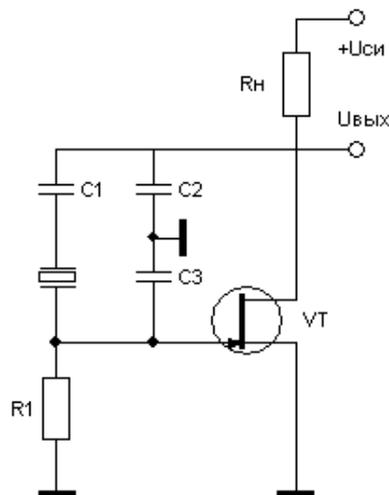


Рисунок 3 – Кварцевый генератор Колпитца

Тактовые кварцевые генераторы находят широчайшее применение в современной технике. Однако производство пьезоэлектрических кварцевых резонаторов трудозатратно поскольку состоит из разнохарактерных технологических процессов с использованием разнообразного оборудования и приборов. Кварцевый элемент, рассчитанный на определенную частоту, имеет определённый размер. Его получают резанием кристалла кварца под установленным углом в зависимости от кристаллической решетки. Сначала элемент шлифуется и полируется, затем осуществляется травление, металлизация, монтаж, настройка, сборка и упаковка в корпус. Производство резонаторов на частоту выше 100 МГц становится сложнее из-за трудностей в изготовлении тонких кварцевых пластин с высокой степенью параллельности рабочих сторон, что сильно влияет на появление резонанса на гармониках вблизи рабочей частоты. Сложность процесса производства приводит к большому количеству отбраковки изделий на разных этапах производства, а также к большим срокам изготовления.

Альтернативным решением для тактовых генераторов стало применение МЭМС резонаторов взамен кварцевых. Они обладают повышенной добротностью, устойчивостью к ударам и вибрациям, стабильностью выходной частоты, меньшими размерами и потребляемой мощностью, позволяют получить более высокую выходную частоту, а также технологически производятся на одной подложке со СБИС. Последний фактор существенно удешевляет серийные МЭМС генераторы.

2 Проектирование резонаторов в системе Coventor MEMS+

Система Coventor MEMS+ представляет собой специализированный программный продукт, предназначенный для проектирования резонатора, от построения концепции проекта до производства, в тесной интеграции с потоком проектирования интегральных микросхем компании Cadence.

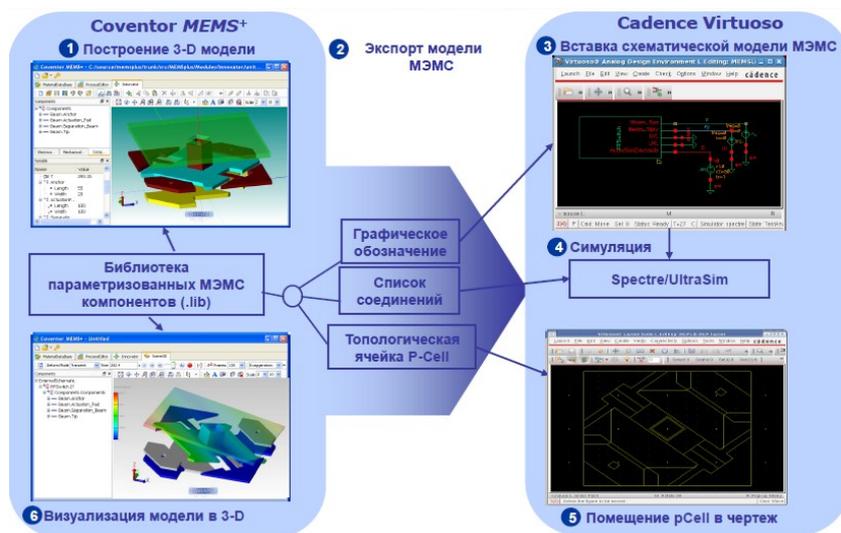


Рисунок 4 – Схема взаимодействия Coventor MEMS+ с Cadence Virtuoso

В основе системы лежит программа INNOVATOR, объединяющая в себе средства визуального проектирования МЭМС устройств, ранее реализованные в пакете CoventorWare. Программа INNOVATOR позволяет построить требуемое МЭМС устройство в виде принципиальной схемы из специальных библиотечных блоков, представляющих собой модели различных элементов МЭМС устройств. Данные модели дают возможность моделировать электрическое и механическое поведение устройства, а также сформировать его трехмерное представление. На любом из этапов проектирования программа INNOVATOR может сформировать данные для программного обеспечения Cadence: библиотеки символов для редактора схем Cadence Virtuoso, списки соединений для мультифизического моделирования с помощью модулей Spectre и UltraSim, а также топологические ячейки P-Cell для редактора топологий микросхем.

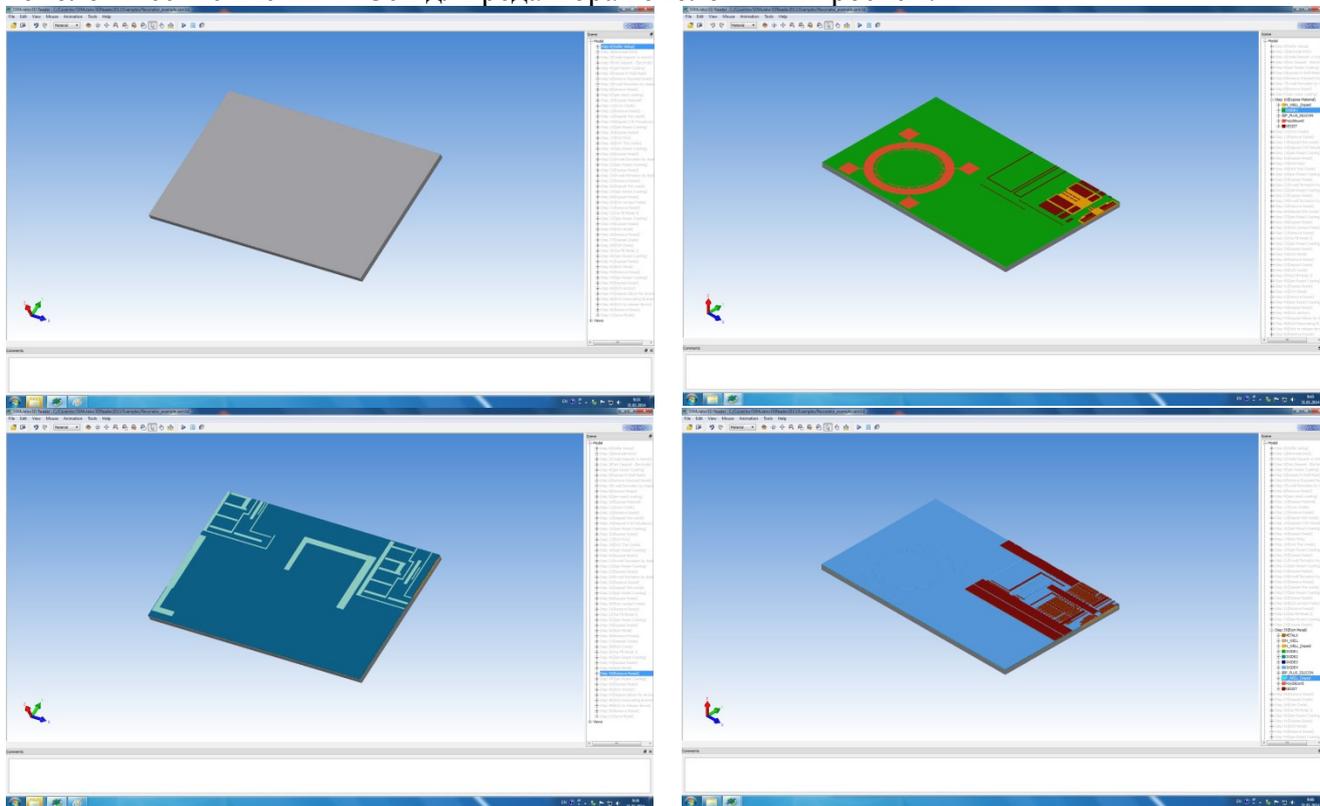


Рисунок 5 – Этапы построения модели МЭМС резонатора в системе Coventor MEMS+

Описанный подход позволяет выполнять моделирование МЭМС устройств в окружении цифровой или аналоговой части микросхемы, ранее спроектированной в пакете Cadence Virtuoso, а также формировать двумерные маски для кристалла в целом.

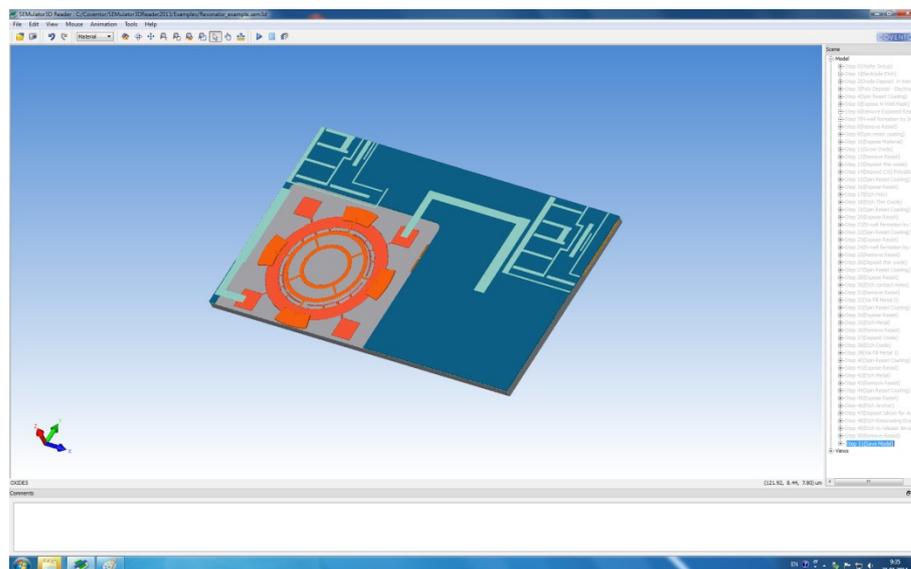


Рисунок 6 – Общий вид модели МЭМС резонатора в системе Coventor MEMS+

Разработанная модель МЭМС резонатора в системе Coventor MEMS+ представлена на рисунках 5-6.

Заключение

Технология МЭМС генераторов имеет большое будущее. Авторы видят продолжение работ в проектировании термокомпенсированных МЭМС генераторов, а также моделировании экспериментов по радиационной стойкости, виброустойчивости и ударопрочности данных генераторов с помощью различных САПР.

Литература

1. Альтшуллер Г.Б., Елфимов Н.Н., Шакулин В.Г. Кварцевые генераторы. - М: Радио и связь, 1984. - 232 с.
2. Joel Hedestig. MEMS based reference oscillator // Linköping – 2005. - 63 p.
3. Макаренко В. Проблемы моделирования кварцевых генераторов в среде NI Multisim // Электронные компоненты и системы - 2010. - №7. - С. 43-49.

АНАЛИЗ ПОДХОДОВ К ПОСТРОЕНИЮ ПОНЯТИЙНОЙ ФАЙЛОВОЙ СИСТЕМЫ

Чернов М.М.

Научный руководитель: к.т.н., доцент, Власов А.И., асс. Аверьянихин А.Е.
МГТУ им. Н.Э.Баумана, кафедра ИУ4, Москва, Россия

ANALYSIS APPROACH TO THE CONSTRUCTION OF CONCEPTUAL FILESYSTEM

Chernov M.M.

Supervisors: Ph.D., associate professor, Vlasov A.I., ass. Averianixin A.E.
MSTU, Moscow, Russia

Аннотация

В статье рассматриваются подходы к организации файловой системы персонального компьютера. Подробно исследованы методы классификации файлов и выявлены их основные недостатки. Кратко представлена концепция понятийной файловой системы, состоящая в разделении классификации файлов на иерархию множеств ресурсов и онтологию предметной области. В заключении представлены рекомендации по использованию концепции понятийной файловой системы для решения проблемы манипулирования единицами информации внутри ресурсов файловой системы.

Abstract

The article discusses approaches to file systems of personal computers. Details are researched methods of classification files and identified their major drawbacks. Briefly introduces the concept conceptual file system, consisting in dividing the classification hierarchy of files on a variety of resources and domain ontologies. In conclusion provides recommendations for using the concept conceptual file system to solve the problem of manipulating bits of information resources within the file system.

Введение

При взаимодействии с вычислительной системой, человек намеренно упрощает свой механизм восприятия информации до уровня идентификации по имени. Реализация этой парадигмы может привести к двум последствиям:

- на устройствах хранения файлов появляется все больше данных, энтропия которых растет и не контролируется;
- на устройствах хранения файлов появляется все больше данных, пользователь тратит все больше времени для контроля разнообразия данных, формируя и поддерживая противоречивую иерархию каталогов файлов.

В первом случае очень высока вероятность дублирования данных, хранящихся в файлах с разными строковыми именами.

При увеличении разнообразия информации, для поддержания структуры каталогов и файлов растет поле атрибутов и отношения между ними. Иерархия каталогов увеличивается «вглубь», формируя файловые системы большой вложенности. В таких условиях растет время, которое необходимо затратить для получения доступа к требуемым данным. Это является следствием намеренного упрощения человеком восприятия информации при взаимодействии с машиной.

1 Организация файловых систем

Для сохранения информации в локальных вычислительных системах на продолжительное время используются системы долговременного хранения данных, например жесткие диски. Для того чтобы получить доступ к данным на жестком диске необходимо либо указать физическое местоположение данных (цилиндр, головку, сектор), либо логическое (номер блока).

Необходимость средств организованного сохранения и выборки данных привела к появлению файловых систем [1]. Операционная система персонального компьютера использует файловую систему для представления данных на жестком диске понятным для пользователя образом.

С точки зрения пользователя, самое важное в файловой системе – это механизм взаимодействия с ней. В существующих файловых системах для управления файлами используется абстракция каталогов. Каталог содержит записи с атрибутами и адресами данных для файлов, находящихся в нем. В простейшем варианте в файловой системе имеется один каталог, в котором хранятся все файлы всех пользователей. Подобные системы были распространены в ранних персональных компьютерах. Виды устройства файловой системы изображены на рисунке 1.

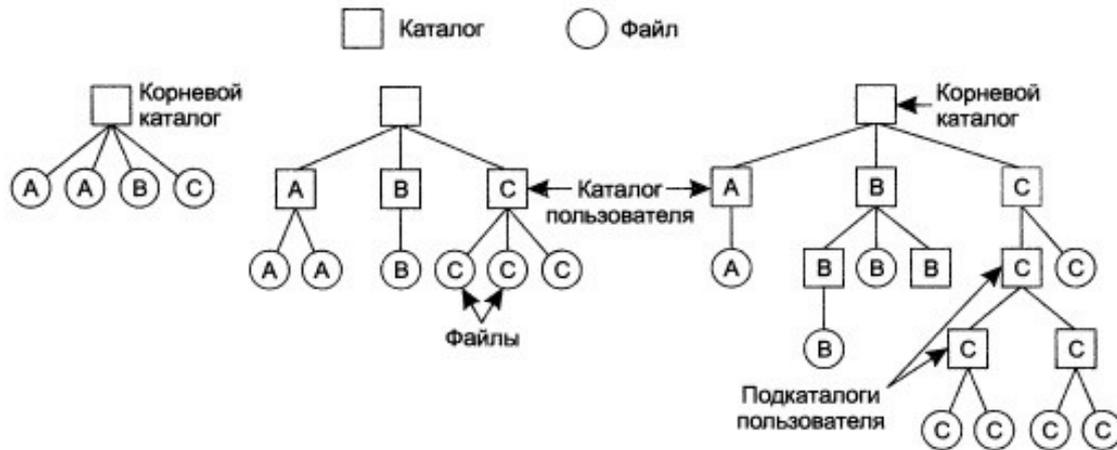


Рисунок 1 Виды устройства файловой системы

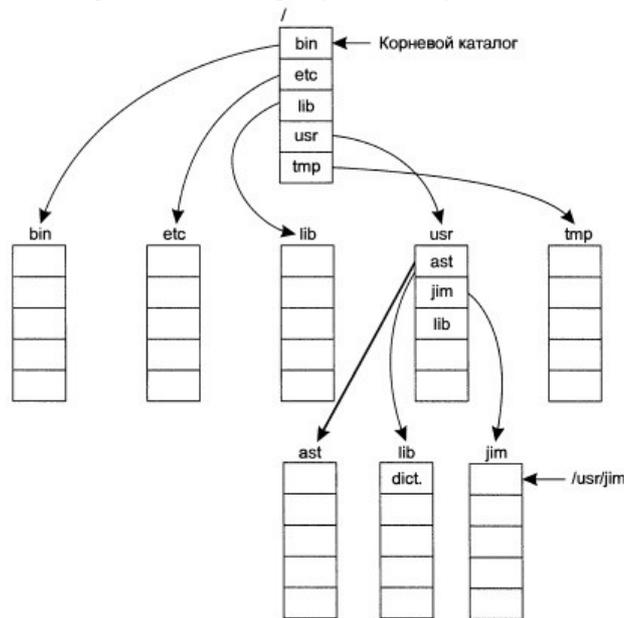


Рисунок 2 Дерево каталогов системы Unix

При этом если пользователей было много, то некоторые из них могли непреднамеренно перезаписать файлы других пользователей. Для предотвращения подобных конфликтов в многопользовательских системах выделялся отдельный каталог под каждого пользователя.

Двухуровневая иерархия помогла исключить конфликты имен между файлами пользователей. Следующий шаг был связан с введением иерархии каталогов. При таком подходе каждый пользователь имеет возможность создавать необходимое число каталогов.

При открытии файла операционная система загружает в оперативную память атрибуты и дисковые адреса открываемого файла [2]. Типовое дерево каталогов операционной системы Unix представлено на рисунке 2.

Таким образом, существующие файловые системы ограничивают варианты организации файлов - заставляют пользователей использовать древовидное расположение файлов и каталогов, причем каталоги, фактически, являются особой разновидностью файлов. Даже если файловая система разрешает использовать другие структуры данных, эта возможность не дается конечным пользователям.

2 Классификация файлов в традиционных операционных системах

Решая задачу классификации данных на персональном компьютере, пользователь создает иерархическую структуру каталогов файлов. При этом классификация получается противоречивой, поскольку один и тот же файл не может находиться в 2-х каталогах одновременно, кроме того, каталогу файлов часто дается название понятия, к которому относятся содержащиеся в нем файлы (например «Микроэлектроника»), меж тем как тип файлов определяется названием родительского (например «Книги») или дочернего (например «Изображения») каталога. Такой подход вынуждает пользователя запоминать, по какому принципу был классифицирован тот или иной ресурс (сначала по типу файлов, а затем – по понятиям, или наоборот) [3].

На рисунке 3 схематично изображена проблема, возникающая из-за неоднородности классификации при использовании абстракции каталогов и предполагаемое решение в концепции понятийной файловой системы



Рисунок 3 – Проблема неоднородности классификаций файлов с использованием абстракции каталогов и предполагаемое решение в концепции понятийной файловой системы

Разделение классификации файлов на дерево множества ресурсов и онтологию предметной области позволит избавиться от неоднородности классификации и даст возможность пользователю обращаться к файлам на своём персональном компьютере через дерево понятий, что для человека предпочтительнее использования абстракции каталогов.

3 Манипулирование единицами логической информации внутри ресурсов файловой системы

Имеется дополнительная сложность доступа к единицам информации, которые находятся внутри файлов. Для доступа к таким единицам пользователю необходимо найти файл, в котором содержится искомая логическая единица информации, а затем осуществить поиск внутри файла.

При этом для доступа к файлу пользователь использует одну из классификаций, в которых содержится искомый файл. Однако такие типы файлов как изображения, аудио- и видео-файлы, документы могут содержать внутри себя огромное количество единиц логической информации и невозможно классифицировать конкретный файл в соответствии с каждой из них. Хотя даже в этом случае пользователь должен будет решить задачу поиска внутри файла, которая в случае аудио- и видео-файлов может занять длительное время.

Решить проблему доступа к таким единицам логической информации можно выделив их из файла где они находятся и классифицировав их как самостоятельные элементы. Построенные при этом классификации будут впоследствии использоваться для доступа к данным ресурсам.

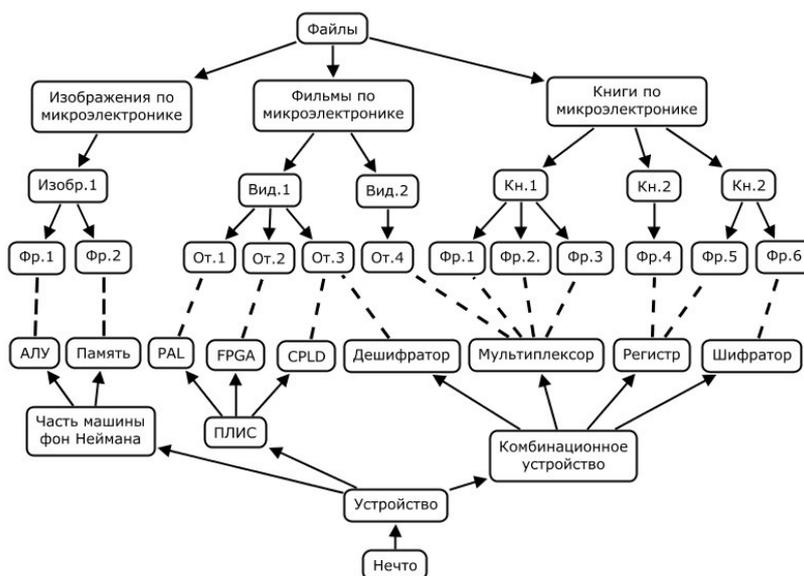


Рисунок 4 – Возможная классификация единиц логической информации находящейся внутри файлов

На рисунке 4 представлена возможная классификация единиц логической информации находящейся внутри файлов.

Использование представленной классификации предпочтительнее использования классификации самих файлов, поскольку она не всегда точно отражает внутреннее содержимое. Кроме того, в этом случае не составляет труда, реализовать машинный поиск логической единицы внутри файла и пользователю не нужно будет тратить свое время.

Заключение

Результатом работы явилась разработка концепции организации файловой системы персонального компьютера путём выделения иерархии множества ресурсов и онтологии предметной области, что позволило решить проблему неоднородности классификации файлов, возникающую при использовании абстракции каталогов.

Предложено решение проблемы манипулирования единицами логической информации внутри ресурсов файловой системы путем включения данных единиц в онтологическую классификацию и создания ссылок на отдельные элементы файлов.

Литература

1. Кэрри Б.Криминалистический анализ файловых систем – СПб.: Питер, 2007. – 480 с.
2. Таненбаум Э., Вудхалл А. Операционные системы. Разработка и реализация – СПб.: Питер, 2007. – 704 с.
3. Чернов М.М., Аверьянихин А.Е., Власов А.И. Использование онтологий для улучшения пользовательского интерфейса персонального компьютера // Научные технологии и интеллектуальные системы - 2013
4. Власов А.И. Гексагональная понятийная модель визуального представления сложных производственных систем// Вестник МГТУ им. Н.Э. Баумана, сер. Приборостроение. 2012. С.157-169.
5. Резчикова Е.В., Власов А.И. Перспективы применения концепт-карт для построения базы знаний ТРИЗ // Сборник трудов конференции "ТРИЗ. Практика применения методических инструментов". - Москва. 29 октября 2011. С.140-145.

МОДЕЛИРОВАНИЕ АВИАЦИОННЫХ РАДИОЛОКАЦИОННЫХ СТАНЦИЙ В РЕЖИМЕ СИНТЕЗИРОВАНИЯ АПЕРТУРЫ С ИСПОЛЬЗОВАНИЕМ ГРАФИЧЕСКИХ ПРОЦЕССОРНЫХ УСТРОЙСТВ

Чернов С. К.

Научный руководитель: д.ф.-м.н., профессор, Карпенко А. П.

МГТУ им. Н.Э.Баумана, кафедра РКБ, Москва, Россия

SIMULATION OF AN AIRBORNE SAR USING GRAPHICS PROCESSING UNITS

Chernov S. K.

Supervisor: Dr., Prof., Karpenko A. P.

MSTU, Moscow, Russia

Аннотация

В статье рассматривается моделирование работы радиолокационной станции в режиме картографирования с использованием графических процессорных устройств (ГПУ). Предложен метод, основанный на стандартной технологии растеризации, позволяющий моделировать РЛС с синтезированной апертурой (РСА). Разработана программа, реализующая данный метод. В заключении приведены перспективы развития метода.

Abstract

The paper describes simulation of radar operating in ground mapping mode using graphics processing unit. A SAR simulation technique based on the standard rasterization technology is presented. A program using the method is designed. Method prospects are shown in the conclusion section.

Введение

Радиолокационные станции (РЛС) применяются для различных целей, в частности, для картографирования поверхности в интересах геологии, топографии, разведки и т.д. При разработке авиационных тренажеров, стендов возникает необходимость моделирования бортовых РЛС в режимах картографирования как с синтезированием апертуры, так и реальным лучом.

Для этого применяют методы компьютерной графики: трассировку лучей, которая позволяет получить более правдоподобный результат [1], и растеризацию (метод сканирования строк), более производительную [2]. Оба метода применяются для моделирования РСА и не позволяют напрямую создать изображения, полученные реальным лучом. От этого недостатка свободно непосредственное моделирование радиосигналов [3]. В данном методе посылается конусообразный луч, близкий к реальному радиолучу. Картографируемая поверхность состоит из множества фасетов, эхо-сигналы от которых суммируются. Однако такой подход требует больших вычислительных затрат.

В данной работе предлагается подход на основе метода бросания лучей (ray casting), близкий к подходу с непосредственным моделированием эхо-сигналов: суммируются отражения от участков поверхности, попавших при растеризации в разные пиксели. При этом можно сократить число фасетов и увеличить их размеры. Такой подход позволит получать более «естественные» по сравнению с растеризацией изображения при сравнимых затратах.

Из-за высокой сложности отладки программ для ГПУ нужно сначала оценить пригодность предложенного подхода к практическому использованию. Такая проверка является целью данной работы. Необходимо решить следующие задачи: разработать метод получения эхо-сигнала в виде, пригодном для синтеза апертуры; выполнить его программную реализацию, использующую преимущественно ЦПУ, проанализировать результат.

В работе дано описание предложенного метода, результатов работы программы. Сделан вывод о целесообразности дальнейшей разработки.

1 Метод моделирования радиосигнала с растеризацией из положения РЛС

В соответствии с методом бросания лучей из точки, в которой находится РЛС (камера) в сцену (набор трехмерных моделей игрового мира) посылаются лучи по числу пикселей создаваемого изображения. Ближайший к камере объект, с которым пересекается луч, определяет цвет пикселя. Данный метод реализован аппаратно во всех ГПУ. Кроме цвета, в процессе отрисовки вычисляется дальность до объекта и нормаль поверхности в месте пересечения с лучом. Сохраним их в видеопамати в виде карты глубины и карты нормалей. В назначаемой объекту текстуре вместо цвета объекта закодируем идентификатор типа его поверхности (например, бетон, асфальт, лес, поле), при этом при отрисовке вместо снимка местности в видимом диапазоне получим карту типов поверхностей.

По картам типов поверхностей и нормалей для каждого пикселя вычислим эффективную поверхность рассеяния (ЭПР) участка поверхности, попавшего при отрисовке в данный пиксель. По дальности, хранящейся в карте глубины, ЭПР, параметрам зондирующего сигнала и носителя найдем параметры эхо-сигнала, соответствующего каждому из пикселей.

Один кадр соответствует зондированию одним импульсом участка поверхности, попадающего в главный лепесток диаграммы направленности антенны. Для моделирования картографирования реальным лучом после растеризации для каждого пикселя полученного изображения вычисляем эхо-сигнал. Все эхо-сигналы, попадающие в один элемент разрешения по дальности, суммируем и записываем в соответствующую этому элементу ячейку. Подробно метод и алгоритм рассмотрен в работе [4].

2 Синтезирование апертуры с использованием исходных данных, полученных методом моделирования радиосигнала с растеризацией из положения РЛС

Для моделирования РЛС бокового обзора с синтезированной апертурой для каждого пикселя преобразуем сигнал в комплексный вид. Для этого эхо-сигнал $u(t)$ умножим на опорные сигналы $u'_{\text{оп}}(t)$ и $u''_{\text{оп}}(t)$, которые сдвинуты по фазе на $\pi/2$ друг относительно друга, а в остальном с точностью до амплитуды совпадают с зондирующим [5]. Зондирующий и эхо-сигналы задаем аналитически, амплитудой, частотой и фазой, сигналы полагаем гармоническими. Умножение сигналов при обработке преобразуем в сложение по известным тригонометрическим формулам. В результате получим сумму сигналов. Программно представим ее в виде массива, в котором хранятся параметры всех сигналов, попавших в определенный канал по дальности. Удалим из данного массива все записи с частотой выше частоты среза, моделируя применение фильтра нижних частот.

Для сдвинутого таким образом в область видеочастот сигнала из аналитически заданных параметров находим отсчет дискретного траекторного сигнала, соответствующий текущему положению РЛС. Сместив РЛС, найдем следующий отсчет. Таким образом, производим накопление траекторного сигнала, состоящего из действительной $u_c(x)$ и мнимой $u_s(x)$ составляющих. Здесь x – координата РЛС (камеры). После накопления требуемого числа отсчетов обработаем их способом быстрой свертки [5], и получим парциальный кадр радиолокационного изображения.

3 Проблема дискретизации траекторного сигнала

В общем случае, частота зондирования выбирается исходя из максимальной частоты траекторного сигнала и теоремы Котельникова. Однако из-за дискретности изображений, служащих источником данных о сигнале, и низкого их разрешения, в сигнал вносятся существенные искажения. Ширина участка поверхности «под пикселем» больше, чем длина траектории, соответствующая периоду колебания сигнала, только тогда, когда цель находится на траверзе и вблизи него. За пределами узкого центрального участка период колебания сигнала существенно уменьшается, что не позволяет его разрешить. В результате картина синтезированного радиолокационного изображения значительно искажается. На рис. 1 показаны траекторный сигнал и реакция РСА на точечную цель с аналитически заданными координатами. На рис. 2 показаны траекторный сигнал и реакция РСА на цель, занимающую в каждый момент времени один пиксель исходного изображения.

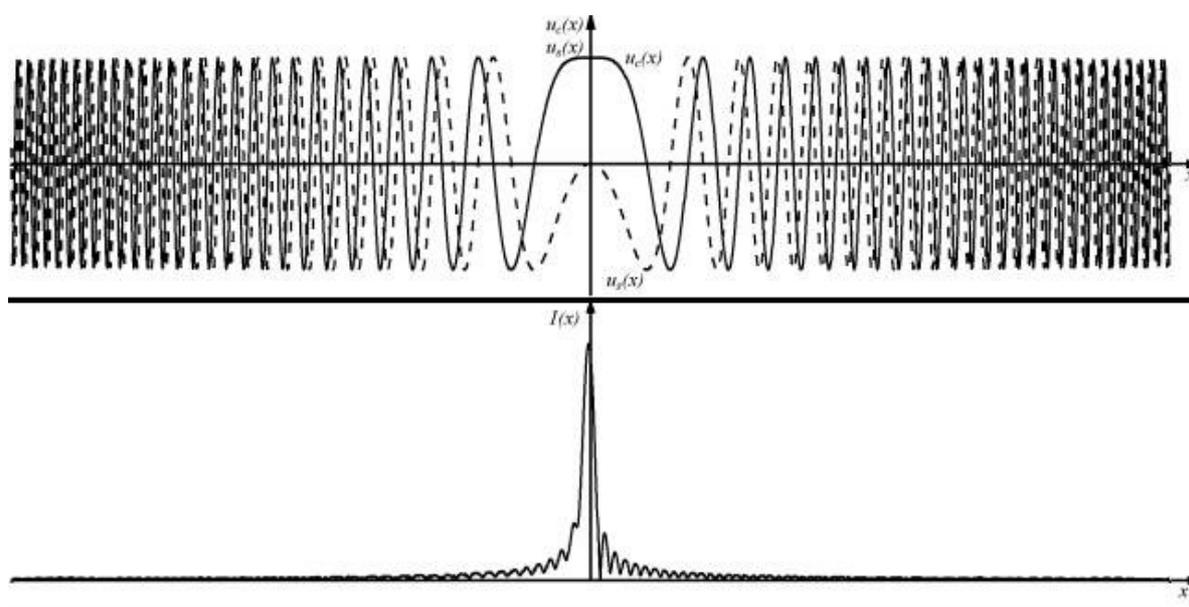


Рисунок 1 — Траекторный сигнал и реакция РСА на точечную цель

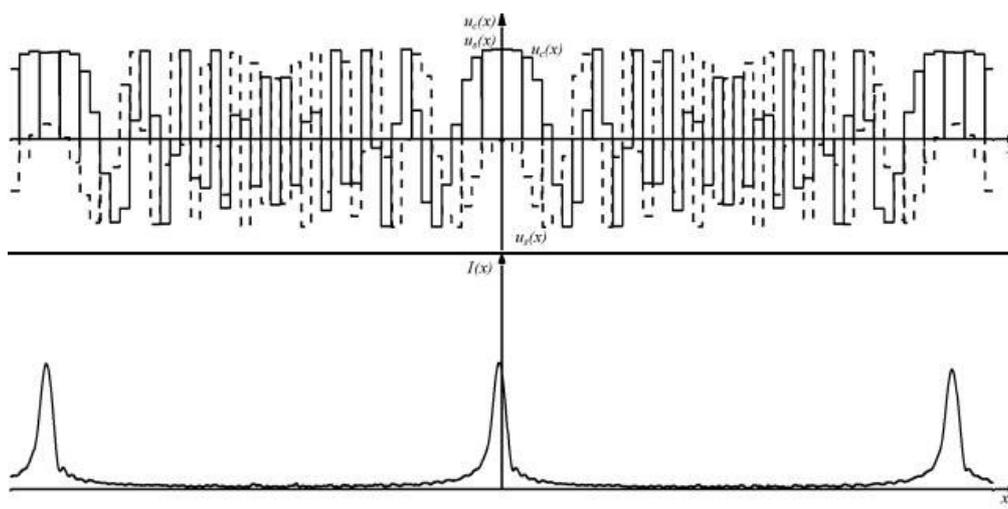


Рисунок 2 — Траекторный сигнал и реакция РСА на цель, дискретизированную методом бросания лучей

Чтобы избежать этих искажений, можно повысить разрешение исходных изображений в десятки раз, однако это потребует и соответствующего повышения затрат

времени для их получения. Поэтому мы предлагаем подход с использованием интерполяции между ключевыми положениями камеры.

При его простейшей реализации сдвиг камеры производится на величину, равную ширине участка поверхности «под пикселем», а перемещение между позициями интерполируется линейно. Однако такой подход применим только если моделируется отражение от одной строки пикселей. Для полноценного изображения сдвиг камеры на наибольший общий делитель «ширин» пикселей всех строк потребует огромных вычислительных затрат. Мы производим сдвиг камеры на фиксированный шаг, а в качестве ключевых позиций, уникальных для каждой строки пикселей, используем те, которые совпадают с идеальной с определенной погрешностью, например, 0,25 «ширины пикселя».

Делаем поправку на погрешность и между двумя соседними скорректированными позициями линейно интерполируем движение камеры.

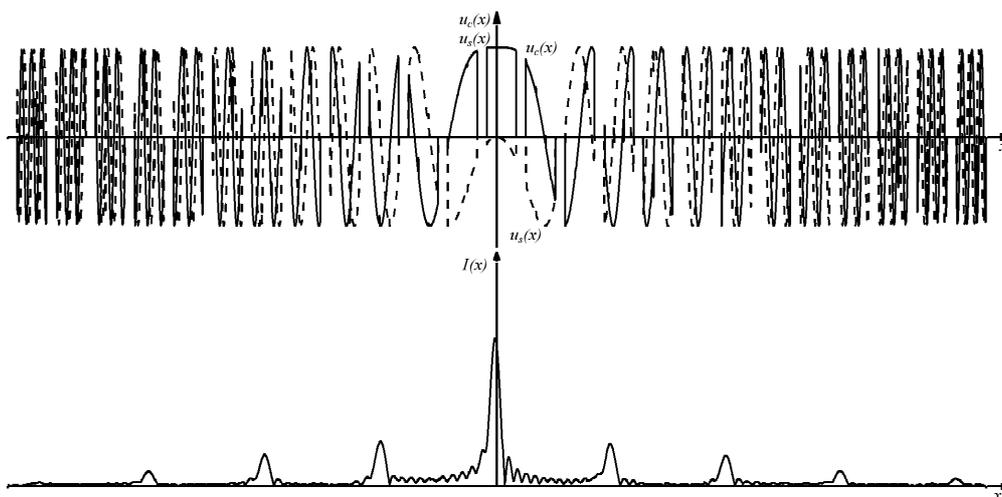


Рисунок 3 — Траекторный сигнал и реакция PCA на цель с координатами, восстановленными интерполяцией между ключевыми кадрами

Из-за погрешности на некоторых ключевых позициях могут, с одной стороны, выпадать цели, видимые на большинстве других, а с другой, есть возможность пропуска цели, однако в целом картина, получаемая с применением этого подхода ближе к картине с аналитически заданным положением цели.

Заключение

Приведенный график (рис. 3), полученный с помощью разработанной программы, показывает, что метод моделирования эхо-сигнала с растеризацией из позиции РЛС в сочетании с интерполяцией положения РЛС между ключевыми кадрами позволяет получить достаточно качественное радиолокационное изображение. Целесообразно продолжить работу над моделью для полного ее переноса на ГПУ.

Литература

1. Auer, S., Hinz, S., Bamler, R., Ray-Tracing Simulation Techniques for Understanding High-Resolution SAR Images. // IEEE Transactions on Geoscience and Remote Sensing — 2010 — С. 1445–1456.
2. Balz, T., Haala, N., Improved Real-Time SAR Simulation in Urban Areas // Proceedings of the IEEE International Geoscience & Remote Sensing Symposium — 2006 — С. 3631–3634.
3. Franceschetti, G., Migliaccio, M., Riccoi, D., Schirinzi, G., SARAS: Asynthetic Aperture Radar (SAR) Raw Signal Simulator // IEEE Transactions on Geoscience and Remote Sensing. — 1992. — № 1. — С. 110-123.
4. Чернов, С. К., Карпенко, А. П., Моделирование авиационных радиолокационных станций в режиме картографирования на графических процессорных устройствах // Международный конгресс по информатике: информационные системы и технологии CSIST'2013: материалы международного конгресса — 2013 — С. 336–340.
5. Радиолокационные станции с цифровым синтезированием апертуры антенны / В. Н. Антипов, В. Т. Горяинов, А. Н. Куклин и др.; под ред. В. Т. Горяинова. — М.: Радио и связь, 1988 — 304 с.: ил.

СПАЙКОВЫЕ НЕЙРОННЫЕ СЕТИ ДЛЯ РЕШЕНИЯ ЗАДАЧ КЛАССИФИКАЦИИ ВРЕМЕННЫХ РЯДОВ

Чернышев А.С.

Научный руководитель: д.ф.-м.н., профессор, Карпенко А.П.

МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

SPIKING NEURAL NETWORKS FOR TIME SERIES CLASSIFICATION TASK

Chernyshev A.S.

Supervisor: Dr., Prof., Karpenko A.P.

MSTU, Moscow, Russia

Аннотация

В статье рассматриваются спайковые нейронные сети в контексте улучшения алгоритмов классификации временных рядов. В качестве базового алгоритма для классификации взят метрический метод минимума квадрата ошибки. Показано, что обученная нейронная сеть может уменьшить уровень ошибки и в целом улучшить метод классификации. В заключении представлены рекомендации по применению спайковых нейронных сетей и рассматривается вектор дальнейших исследований по данной теме.

Abstract

Spiking neural networks in the context of improvement of time series classification algorithms are the object of this article. As a baseline for classification we take metric method of square error minimum. It is shown that trained neural network can diminish the error level and enhance statistic properties of classification method. In conclusion, recommendations for spiking neural networks usage are presented and direction of further research on given topic are examined.

Введение

Задача классификации временных рядов является одной из классических задач компьютерных наук и машинного обучения. Задачу классификации можно определить, как создание алгоритмов способных определить, к какому классу принадлежит тот или иной временной ряд тестовой выборки, исходя из особенностей тренировочной выборки. Существует набор данных, созданный в University of California [1], который включает в себя 40 наборов данных и является «лакмусовой бумажкой» любого алгоритма классификации одномерных временных рядов.

Актуальной является задача улучшения классических алгоритмов классификации, а также создание новых, более точных. В данной работе рассматривается метод обучения спайковых нейронных сетей без учителя, который позволяет обучить сеть на тренировочных данных и получить на выходе преобразованный временной ряд в виде спайков, который является «эссенцией» исходного временного ряда и несет в себе необходимые характеристики, чтобы повысить эффективность алгоритма классификации. В качестве исходного алгоритма классификации выбран наиболее простой алгоритм – метрический метод классификации на основе минимума суммарного квадрата разницы между известным тренировочным временным рядом и тестируемым.

Данная работа является логическим продолжением работ [2 - 4], в которых установлены базовые алгоритмы обучения спайковых нейронных сетей с учителем. Была произведена необходимая модификация алгоритмов для возможности обучения генеративного характера, т.е. без учителя.

1 Спайковые нейронные сети

Нейрон в спайковых нейронных сетях представляет собой динамическую систему, которая преобразует входную последовательность спайков с M синапсов в выходную

последовательность спайков. Спайком является событие, которое характеризуется временем, когда оно произошло, и нейроном, который его произвёл. Каждый нейрон имеет внутреннее состояние, которое зависит от взвешенной суммы спайков с входных синапсов, пройденных через операцию свёртки с функцией постсинаптического потенциала, а также от недавней активности нейрона – нейрон имеет состояние рефракторности, которое уменьшает вероятность появления нового спайка, если нейрон их недавно генерировал. Вероятность спайка выражается плотностью негомогенного пуассановского процесса, которая зависит от состояния нейрона.

Соединение таких нейронов в рекуррентные сети позволяет получить выход сети, который сочетает в себе как характеристики входного сигнала, так и результат нелинейной обработки сигнала нейронами.

Симуляция такого рода сетей производится путем подсчёта состояний каждого нейрона в дискретные интервалы времени.

2 Постановка задачи

Требуется обучить спайковую нейронную сеть так, чтобы на каждый входной временной ряд сеть генерировала уникальный выход, который лучшим образом описывает входные данные. Выход сети должен быть результатом обработки большой выборки данных и должен отражать его статистические особенности.

Вопрос улучшения метода классификации в контексте поставленной задачи играет ключевую роль. Предложенный метод должен быть использован для получения опорного рейтинга ошибки, без применения нейросетевой модели. После чего метод должен быть применен к обработанным выходным сигналам сети для подсчёта рейтинга, который даёт предложенная модель.

3 Правило обучения

В качестве правила обучения нейронной сети взят принцип повышения правдоподобия каждого выходного спайка сети в ответ на любую входную последовательность. Таким образом, сеть обучается генерировать всё более устойчивый ответ на один и тот же вход и энтропия сети уменьшается. Более подробно фреймворк описан в работах [2 - 4].

Ввиду нового способа применения такого правила обучения, в ходе вычислительных экспериментов были обнаружены их новые свойства. При отсутствии сдерживающего фактора в сети, который бы ограничивал активность, повышение правдоподобия всех спайков сети со временем приводит к гиперактивности – сеть выдает на разные входные данные одну и ту же осцилляционную активность, которая только повышает рейтинг ошибки алгоритма классификации. Таким образом, принцип обучения был изменен: при достижении активности всей сети выше пороговой, функция правдоподобия начинала учитывать только те спайки, вероятность, которых была выше средней по нейрону.

4 Вычислительный эксперимент

В качестве входных данных взят первый набор данных из комплекта UCR [1] – Synthetic control, который состоит из 300 тренировочных и 300 тестовых временных рядов 6 разных классов длиной в 60 измерений. В качестве алгоритма классификации использован метод минимума квадрата расстояния между двумя рядами.

Вопрос представления этих данных в сети был решен созданием 50 входных нейронов, каждый из которых ответственен за ту или иную область измерений во временном ряде. Входные нейроны последовательно один за другим генерируют спайки в зависимости от текущего положения временного ряда в ходе симуляции. Каждая группа входных

нейронов соединяется с одним нейроном сети, так чтобы число входных соединений на нейрон было равномерное.

Нейроны сети соединены друг с другом с вероятностью 15% с исключением связей с самим собой.

Длительность симуляции взята из расчёта длины постсинаптической функции – нейроны должны успевать реагировать на входную последовательность.

Более детально параметры симуляции рассмотрены в таблице 1.

5 Применение сети

После обучения, тренировочный и тестовый наборы подаются на вход нейронной сети и обрабатываются соответствующие ответы. Ответ сети представляет собой набор времен спайков нейронов сети и малопригоден для сравнения ответов между собой, поэтому ответы были разделены на участки и проинтегрированы ядром гауссовского профиля с одним параметром. Таким образом, на выходе сети получается матрица, количество строк которой равно количеству нейронов сети, количество колонок – количеству проинтегрированных участков.

Ввиду стохастичности сети, существует необходимость взять среднее от ответа для каждого временного ряда. Число необходимых итераций и параметр ядра устанавливался по принципу минимума допустимой ошибки. Обработанные ответы подавались на вход алгоритма классификации.

Таблица 1 - Параметры эксперимента. $H(s)$ -функция Хэвисайда

Параметр	Значение
Постсинаптическая функция	$f(s) = e_0 (e^{-\frac{s}{t_m}} - e^{-\frac{s}{t_s}}) H(s); e_0 = 1.3, t_m = 10, t_s = 1'$
Функция рефракторности	$f(s) = (U_{abs} e^{-\frac{s+d_r}{t_{rf}}} - U_r e^{-\frac{s}{t_{rs}}}) H(s);$ $U_{abs} = -150, t_{rf} = 3.25, d_r = 1, t_{rs} = 3;$
Стохастичный порог	$f(u) = \frac{\beta}{\alpha} \log(1 + e^{\alpha(U_{tr} - u)}) - \alpha(U_{tr} - u); U_{tr} = -50, \beta = 0.7$
Количество входных нейронов	50 нейронов
Размер групп входных нейронов	6 нейронов граничные группы; 7 нейронов - остальные
Количество нейронов сети	10 нейронов
Вероятность соединения внутри сети	15%
Длительность симуляции	100 мс
Количество эпох	27
Коэффициент обучения	0.01
Пороговая активность	45 спайков
Ядро	$K(t, \Omega) = \sum_{t_f \in \Omega} \frac{1}{\sqrt{2\pi}} e^{-\frac{(t-t_f)^2}{2\sigma^2}}, \Omega \text{ - набор точек}, \sigma^2$
Количество проинтегрированных участков	8
Количество итераций усреднения	10

6 Анализ результатов

В таблицах 2, 3 представлен результат решения поставленной задачи в виде матриц неточностей классификаторов без участия нейросетевой модели и с участием соответственно.

Таблица 2 – Матрица неточностей классификатора

	1	2	3	4	5	6
1	22	10	3	4	5	6
2	0	40	0	0	0	0
3	0	0	49	0	1	0
4	0	0	0	50	0	0
5	0	0	3	0	47	0
6	0	0	0	4	0	46

Таблица 3 – Матрица неточностей классификатора на базе нейронной сети

	1	2	3	4	5	6
1	36	10	1	2	0	1
2	2	46	0	2	0	0
3	0	0	46	0	4	0
4	0	0	0	50	0	0
5	0	0	6	0	44	0
6	0	0	0	1	0	49

Видно, что алгоритм классификации стал показывать лучший результат по сравнению с исходным рейтингом ошибки классификатора: 9.66% ошибок против 12%.

Заключение

Данная работа демонстрирует лишь некоторые возможности спайковых нейронных сетей. Не смотря на то, что текущее решение очень громоздко и не оптимально, оно является базовым, отталкиваясь от которого возможны усовершенствования, например, оптимизация параметров сети и числа нейронов, усложнения процедуры постобработки ответов, усложнение алгоритма классификации ответов.

Литература

1. Keogh, E., Zhu, Q., Hu, B., Hao, Y., Xi, X., Wei, L. and Ratanamahatana, C. A. (2011). The UCR Time Series Classification/Clustering Homepage: www.cs.ucr.edu/~eamonn/time_series_data/
2. T. Toyozumi, J.-P. Pfister, K. Aihara and W. Gerstner, Optimality Model of Unsupervised Spike-Timing Dependent Plasticity: Synaptic Memory and Weight Distribution — Neural Computation, 19, 2007 — 639-671 p.
3. J.-P. Pfister, T. Toyozumi, D. Barber and W. Gerstner, Optimal Spike-Timing Dependent Plasticity for Precise Action Potential Firing in Supervised Learning — Neural Computation, 18, 2006 — 1309-1339 p.
4. Bohte, S.M., Mozer, M. C., Reducing the variability of neural responses: A computational theory of spike-timing dependent plasticity – Neural Computation, 19, 2009. – 371-403 p.

ОБЗОР МЕТОДА РЕЛАКСАЦИИ ДЛЯ ПОИСКА ТОЧЕК РАВНОВЕСИЯ ПО НЭШУ В НЕПРЕРЫВНЫХ НЕКООПЕРАТИВНЫХ ИГРАХ МНОГИХ ЛИЦ

Чиркина Д.Н.

Научный руководитель: д.т.н., профессор, Карпенко А.П.

МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

REVIEW ON RELAXATION METHOD OF SEEKING THE NASH EQUILIBRIA IN NONCOOPERATIVE CONTINUOUS GAMES OF MANY PERSON

Chirkina D.N.

Supervisor: Dr., Prof., Karpenko A.P.

MSTU, Moscow, Russia

Аннотация

В статье рассматривается задача поиска точек равновесия по Нэшу в непрерывных некооперативных играх многих лиц. Кратко приводится формулировка принципа равновесия по Нэшу. Подробно рассмотрены метод релаксации Нэша. В заключении производится анализ его эффективности.

Abstract

This paper devoted to Seeking the Nash Equilibria in noncooperative continuous games of many persons. The formulation of the Nash Equilibria is shortly given in the paper. Nash Relaxation method of Seeking the Nash Equilibria have been considered in the paper. Analysis of the method has been represented in conclusion.

Введение

Задача теории игр известны своей сложностью. Однако необходимость поиска точек равновесия по Нэшу возникает во многих экономических и инженерных задачах.

Принцип равновесия по Нэшу был сформулирован в 1950 г. в работе Дж. Нэша «Точки равновесия в играх n лиц» [1]. Данный принцип является одним из ключевых понятий теории игр.

В настоящее время существует значительное количество методов для решения задачи поиска точек равновесия по Нэшу, среди которых методы Никайдо и Изода, Розена, Смейла, методы частично-целочисленного программирования и другие.

В данной работе мы рассматриваем и анализируем эффективность метода релаксации Нэша. Данный метод, являющийся модификацией широкоизвестного метода релаксации, был разработан в 1999 Джасеком Б. Краусцеком, ученым из Университета Виктории (г. Веллингтон, Новая Зеландия), и Станиславом Ярясевым из Университета Флориды (США) [2].

1 Постановка задачи

Пусть имеется n игроков, взаимодействующих друг с другом. Каждый игрок i из этих n игроков располагает набором возможных альтернативных стратегий X_i . Каждый игрок i может выбрать действие $x_i \in X_i$. Совокупность действий всех игроков – вектор $\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{R}^{k_1} \times \dots \times \mathbb{R}^{k_n}$, где k_i – размерность вектора \mathbf{x} . Множество (возможных) коллективных действий или множество ситуаций $X \subseteq X_1 \times \dots \times X_n \subseteq \mathbb{R}^{k_1} \times \dots \times \mathbb{R}^{k_n} = \mathbb{R}^k$ – декартово произведение всех возможных стратегий игроков, отображающееся на множество действительных чисел, и $\mathbf{x} \in X$.

2 Метод релаксации Нэша

Пусть $\mathbf{x} = (x_1, \dots, x_n)$, $\mathbf{y} = (y_1, \dots, y_n) \in X_1 \times \dots \times X_n$, а вектор $(y_i | \mathbf{x}) = (x_1, \dots, y_i, \dots, x_n)$ – такой вектор коллективных действий, при котором игрок i выбирает действие y_i , в то время как другие игроки продолжают выбирать действия из вектора \mathbf{x} .

Введем в рассмотрение оптимальную функцию отклика (функцию ответа). Это функция лучшего ответного действия игрока 1 на действия игрока 2, т.е. функция, которая максимизирует функцию платы игрока 1.

В работе Краусцека и Ярясева показано, что алгоритм сходится к точке равновесия по Нэшу, если таковая существует, при $k \rightarrow \infty$, и данная точка является точкой нормализованного равновесия. Для её существования (а, следовательно, и для сходимости алгоритма) должны выполняться необходимые условия:

- 1) X есть выпуклое компактное подмножество \mathbb{R}^m ;
- 2) функция Никайдо и Изода слабо выпукло-вогнутая: $\Psi: X \times X \rightarrow \mathbb{R}$ и $\Psi(\mathbf{x}, \mathbf{x}) = 0$ для $\mathbf{x} \in X$;
- 3) $Z(\mathbf{x})$ однозначна и непрерывна на X ;
- 4) выбран оптимальный шаг $\alpha_k = \alpha_k^*$ согласно формуле.

Данный метод является методом нулевого порядка (прямым методом), одношаговым, итерационным, детерминированным и последовательным. В качестве условий окончания итераций используются стандартные условия окончания поиска.

3 Результат работы алгоритма на типовом примере

Была создана программная реализация описанного метода релаксации Нэша на языке программирования C++. Для исследования эффективности метода и его программной реализации использован следующий пример.

Имеем двух игроков с одинаковыми функциями платы $q_1(\mathbf{x}) = q_2(\mathbf{x})$, определенными на области $X = \{(x_1, x_2): -10 \leq x_1, x_2 \leq 10\}$, ценовые функции которых имеют вид $q_i(\mathbf{x}) = -\frac{(x_1 + x_2)^2}{4} - \frac{(x_1 - x_2)^2}{9}$, $i = 1, 2$ (рис. 1).

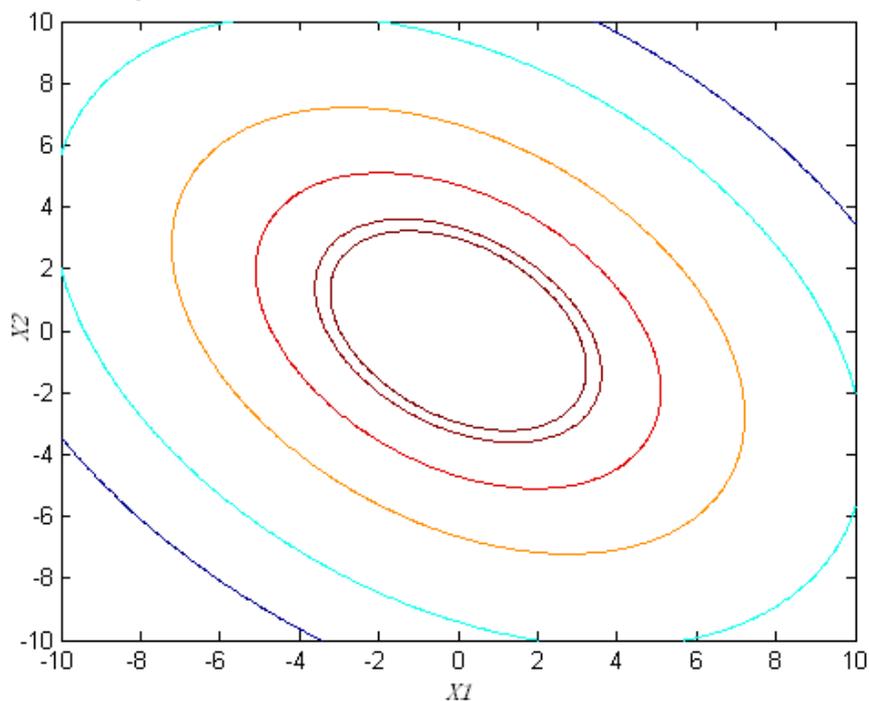


Рисунок 1 – Линии уровня ценовых функций $q_1(x_1, x_2) = q_2(x_1, x_2)$

Предварительно вычислим точку равновесия по Нэшу аналитически:

$$\frac{\partial q_i(x)}{\partial x_i^{(i)}} = \begin{pmatrix} -\frac{1}{2} x_1(x_1 + x_2) - \frac{2}{9} x_1(x_1 - x_2) = -\frac{1}{18} x_1(13x_1 + 5x_2) \\ -\frac{1}{2} x_2(x_1 + x_2) + \frac{2}{9} x_2(x_1 - x_2) = -\frac{1}{18} x_2(5x_1 + 13x_2) \end{pmatrix} = 0, \quad i = 1,2.$$

Следовательно, точкой равновесия является решение $(x_1, x_2) = (x_1^*, x_2^*) = (0; 0)$.

Функция Никайдо и Изода для рассматриваемой задачи имеет вид $\Psi(x, y) = 2 \cdot \left(\frac{(x_1 + x_2)^2}{4} + \frac{(x_1 - x_2)^2}{9} \right) - \left(\frac{(y_1 + x_2)^2}{4} + \frac{(y_1 - x_2)^2}{9} \right) - \left(\frac{(x_1 + y_2)^2}{4} + \frac{(x_1 - y_2)^2}{9} \right)$.

Функция оптимального отклика определяется выражением $Z(x) = -\frac{5}{13}(x_1, x_2)$.
Ценовая функция для k -й итерации вычисляется по формуле $q_i(x^k) = q_i\left((1 - \alpha_k^*) \cdot x^k + \alpha_k^* \cdot Z(x^k)\right), k = \{0, 1, 2, \dots\}$.

Зададимся точностью $\varepsilon = 0,001$ и воспользуемся методом мултистарта. В качестве критерия окончания итераций используем условие $\max(x_1, x_2) \leq \varepsilon$, где $\max(x_1, x_2)$ – норма вектора (x_1, x_2) .

В результате работы программы были получены следующие данные.

Таблица 1 – Решения задачи, полученные методом мултистарта

Начальные приближения, (x_1^0, x_2^0)	Решение, (x_1^*, x_2^*)	Количество вычислений ценовой функции, N
(0,4;0,4)	(0,000994533; 0,000994533)	108
(2;1)	(0,000906079; 0,000453039)	122
(4;2)	(0,000934789; 0,000467395)	127
(6;3)	(0,000929723; 0,000464862)	130
(8;4)	(0,000937198; 0,000468599)	132
(10;5)	(0,000881618; 0,000440809)	134
(400;140)	(0,000959976; 0,000335992)	157

Исследование показало, что в случае отказа от применения оптимального шага в пользу постоянного шага, скорость сходимости алгоритма значительно снижается. Например, при входных параметрах $(x_1^0, x_2^0) = (0,4; 0,4)$ и постоянном значении шага значения решения $(x_1^*, x_2^*) = (0,000999468; 0,000999468)$, удовлетворяющее заданной точности $\varepsilon = 0,001$, достигается за $N = 11977$ вычислений ценовой функции.

Заключение

Методы поиска равновесия по Нэшу предоставляют удобный инструмент для вычислений в широком спектре инженерных и экономических задач. Поскольку в настоящее время подобных методов существует значительное количество, является актуальной проблема их сравнения, анализа и совершенствования.

Рассмотренный в работе метод релаксации Нэша с оптимальным шагом представляет собой пример рационального усовершенствования метода релаксации. На основе метода была создана программа, результаты работы которой проиллюстрированы на примере. Было проведено исследование эффективности метода, результаты отражены в данной работе.

Преимуществами метода релаксации Нэша являются простота реализации алгоритма, применимость для решения задачи поиска равновесия в непрерывных играх, в том числе многопользовательских некооперативных играх, составляющих внушительную долю задач теории игр, и значительно более высокая скорость сходимости в сравнении со скоростью метода с постоянным шагом. Недостатками метода являются невозможность использования параллельных вычислений при реализации его алгоритма, а также необходимость иметь аналитические выражения для функций.

Литература

1. J.F. Nash. Equilibrium points in n -person games // Proc. of the National Academy of Sciences 36, USA, 1950 – pp. 48–49.
2. Jacek B. Krawczyk and Stanislav Uryasev. Relaxation algorithms to find Nash equilibria with economic applications // Environmental Modeling and Assessment 5 (2000) 63–73. – Baltzer Science Publishers BV, 2000 – pp. 63–73.
3. Srihari Govindan and Robert Wilson. A Global Newton Method to Compute Nash Equilibria // Stanford Business School, Stanford, CA and Department of Economics, The University of Western Ontario, London, Ontario, Canada / Social Sciences and Humanities Research Council of Canada and the National Science Foundation of the United States, 1998 – 2001. JEL Classification C63.
4. J.B. Rosen. Existence and uniqueness of equilibrium points for concave n -person games // Econometrica 33(3), 1965 – pp.520–534.
5. Tuomas Sandholm, Andrew Gilpin, Vincent Conitzer. Mixed-Integer Programming Methods for Finding Nash Equilibria // National Science Foundation under ITR grants and Sloan Fellowship / AAAI-05: American Association for Artificial Intelligence, 2005 – pp. 495–501.
6. H. Nikaido, K. Isoda. Note on noncooperative convex games // Pacific Journal of Math. 5 (Suppl. I), 1955 – pp. 807–815.
7. D. Gabay, H. Moulin. On the uniqueness and stability of Nash-equilibria in noncooperative games // In: Bensoussan, A., Kleindorfer, P., Tapiero, C. (eds.) Applied Stochastic Control in Econometrics and Management Science, North Holland, Amsterdam, 1980 – pp. 271–293.

НЕЙРОННЫЕ СЕТИ В ЗАДАЧАХ КОМПЛЕКСНОЙ ОБРАБОТКИ ИНФОРМАЦИИ

Дзюев А.А.^{2,3}

*Научные руководители: Бабиченко А.В.^{1,3} – доктор технических наук,
Некрасов А.В.^{2,3} – кандидат технических наук*

ОАО «Раменское приборостроительное конструкторское бюро» (1)

ЗАО «Инерциальные технологии «Технокомплекса» (2)

МГТУ им. Н.Э.Баумана, кафедра РК6, Москва, Россия (3)

NEURAL NETWORK IS A PROBLEM IN COMPLEX OF INFORMATION PROCESSING

Dzuev A.A. (2, 3)

*Supervisors: Babichenko A.V. (1,3) - doctor of technical sciences,
Nekrasov A.V. (2, 3) - Ph.D.*

JSC "Design office in Ramensk" (1)

JSC "Inertial Technology of "Technokomplex"(2)

Moscow State University n.a. N.E. Bauman, Department RK6, Moscow, Russia (3)

Аннотация

В работе рассмотрено программно-математическое обеспечение современных навигационных комплексов, которое включает многоуровневую систему комплексной обработки информации (КОИ). Одной из задач алгоритмов КОИ является идентификация погрешностей блока чувствительных элементов (БЧЭ) и навигационных систем, в том числе инерциальных - ИНС. В качестве основного метода идентификации погрешностей используются модификации фильтра Калмана, имеющие ряд существенных ограничений, связанных с размерностью модели состояния объекта, наблюдаемостью и последовательной сходимостью оцениваемых параметров, точностью и устойчивостью прогнозного решения.

Abstract

This paper is considered of software of modern navigation systems, which includes tiered complex information processing (CIP). One of the objectives of algorithms CIP is identification of errors of sensor elements unit (SEU) and navigation systems, including inertial - INS. As the main method of identification errors modification is modifications of the Kalman filter with a number of limitations which are related to the dimension of the model object state observability and consistent convergence of the estimated parameters, precision and stability of predictive solutions.

Введение

Указанные ограничения могут быть устранены посредством включения в состав алгоритмов КОИ математических моделей искусственных нейронных сетей (НС), задачей которых является – учет нелинейных составляющих моделей погрешностей. Синтез алгоритмов КОИ и нейросетевых алгоритмов может позволить существенно повысить качество оцениваемых параметров [1-3].

1 Проблема формирования обучающего множества

Ключевой задачей подготовки нейросетевых алгоритмов является формирование обучающего множества, которое включает наборы входных (измерения) и выходных данных (эталон). В процессе обучения на вход НС подаются значения обучающей выборки. Посредством алгоритма обучения устанавливается соответствие между выходами НС и заданным обучающим множеством (рис. 1). Обученная таким образом НС в режиме работы формирует выходные значения в соответствии с заданным обучающим множеством.

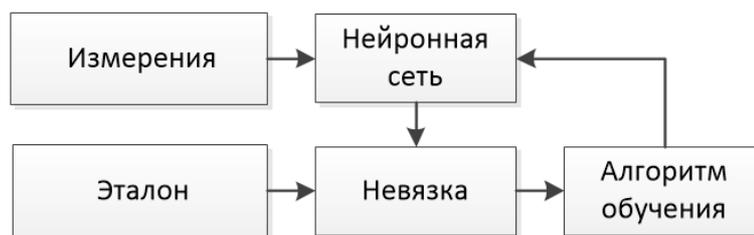


Рисунок 1 - Схема обучения НС

Примером использования НС в алгоритмах КОИ является решение задачи идентификации измерительных погрешностей инерциальной навигационной системы. Основой этих алгоритмов является теория фильтра Калмана, позволяющая определить погрешности системы при наличии корректирующей информации и описать поведение системы при её отсутствии. Изменение погрешностей системы при работе фильтра Калмана в режиме прогноза описывается с помощью следующих соотношений:

$$\dot{X}(t) = A(t) X(t) + G(t) W(t),$$

где $A(t)$ – матрица объекта (состояния) системы; $X(t)$ – вектор состояния; $G(t)$ – матрица возмущения; $W(t)$ – вектор возмущения.

Эти уравнения не учитывают нелинейность поведения системы, не позволяют отслеживать временные изменения погрешностей измерительных датчиков, что приводит к возрастанию со временем погрешностей определения основных параметров ориентации и навигации системы.

С целью уменьшения ошибок прогнозирования предлагается дополнить уравнения фильтра Калмана нелинейной составляющей, формируемой искусственной нейронной сетью по данным обучающей выборки:

$$\dot{X}(t) = A(t) X(t) + G(t) W(t) + f(t)$$

Для формирования обучающей выборки предлагается использовать полученные при математическом моделировании полёта по заданной траектории и лётных испытаниях данные ИНС, спутниковой навигационной системы (СНС) и оценки погрешностей ИНС при работе фильтра Калмана в режиме оценивания. Структурная схема для формирования обучающей выборки представлена на рисунке 2. В блоке 1 содержится база знаний – наборы данных обучающей выборки, включающие рассчитываемые ИНС навигационные параметры, параметры ориентации, измерительную информацию БЧЭ и корректирующие данные от СНС.



Рисунок 2 - Формирование обучающей выборки для НС

В блоке 2 определяются информационные невязки фильтра Калмана, рассчитываемые как разность показаний ИНС и СНС. С учетом информационных невязок и данных от БЧЭ, в блоке 3 производится оценивание параметров вектора состояния, которые используются в блоке 4 для коррекции показаний ИНС. В блоке 5 рассчитывается разность между скорректированными показаниями ИНС и СНС, используемая для формирования обучающей выборки. Полученные значения используются в блоке 6 для обучения НС.

2 Особенности обучения нейронной сети

В результате обучения нейронная сеть формирует составляющие для уточнения модели погрешностей системы в соответствие с измеряемыми значениями ускорений, угловых скоростей, горизонтальной и вертикальной составляющих путевой скорости объекта и его высоты:

$$f_{нл}(t) = f_{нс}(A, W, H, V).$$

Схема использования нейронной сети в режиме работы представлена на рисунке 3.



Рисунок 3 - Схема использования НС

Измерения навигационной системы подаются на вход нейронной сети, которая формирует значения для уточнения модели погрешностей. Полученные значения используются в уравнениях прогноза фильтра Калмана.

Заключение

Подобные комплексные алгоритмы войдут в состав системы КОИ перспективных навигационных комплексов ЛА.

Литература

1. Орехов М.И., Бабиченко А.В., Сухоруков С.Я., Шкред В.К. Многоуровневая система базовых алгоритмов комплексной обработки информации // Инженерная физика. 2012. - №12.
2. Бабиченко А.В., Некрасов А.В., Алексеев А.Н. Математические модели нейронных сетей в задачах бортовых приборных комплексов // Авиакосмическое приборостроение, 2012 г. - № 2.
3. Джанджгава Г.И., Бабиченко А.В., Требухов А.В., Некрасов А.В. Нейросетевой алгоритм калибровки волнового твердотельного гироскопа // Инженерная физика, 2010. -№9.

СОЗДАНИЕ КОНТЕНТНОГО ФИЛЬТРА СЕТИ ИНТЕРНЕТ НА ОСНОВЕ ОТКРЫТОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Хаперская В.Ю.

Научный руководитель Троицкий И. И.

МГТУ им. Н.Э.Баумана, кафедра ИУ8, Москва, Россия

DEVELOPMENT OF INTERNET CONTENT FILTERING SYSTEM BASED ON OPEN-SOURCE SOFTWARE

Haperskaya V.Y.

Supervisor: Troitshkyi I.I.

Bauman MSTU, Moscow, Russia

Аннотация

В работе представлена система контентной фильтрации ресурсов сети Интернет, к которой подключено большинство средних школ в рамках национального проекта “Образование”. Система обеспечивает защиту школьников от информации, пропаганды и агитации, наносящих вред их здоровью, нравственному и духовному развитию.

Abstract

In this project I introduce an Internet content filtering system. Under the national project “Education” most secondary schools were connected to the Internet. This system protects students from information and propaganda, that could be harmful for their health and moral development.

Введение

В настоящее время в Российской Федерации большинство средних школ подключено к сети интернет в рамках национального проекта “Образование”. Интернет является прогрессивным и удобным инструментом обучения, но при этом помимо полезной информации содержит ресурсы экстремистского характера и другой нежелательный контент для детей школьного возраста. Федеральный закон «Об основных гарантиях прав ребёнка в Российской Федерации» предусматривает защиту ребёнка от информации, пропаганды и агитации, наносящих вред его здоровью, нравственному и духовному развитию [1]. Согласно данному закону государственные органы обязаны принять меры по организации такой защиты. Таким образом, в настоящее время в школах особенно остро стоит проблема внедрения системы контентной фильтрации ресурсов сети Интернет, поэтому задача по разработке такой системы является необходимой.

Актуальность работы. Школьные образовательные учреждения в большинстве случаев не обладают необходимыми денежными ресурсами для закупки таких сложных программных комплексов, поэтому целесообразно для создания контентного фильтра использовать свободное программное обеспечение на базе операционной системы GNU/Linux.

Объектом исследования является операционная система GNU/Linux, которая предлагается на рынке бесплатно, но, главное, GNU/Linux зарекомендовала себя как надёжная и стабильная операционная система, не требующая постоянного обслуживания [2]. Для данной операционной системы существуют все необходимые программы для создания контентного фильтра. Данные программы также бесплатны и могут использоваться в любых целях на основе свободной лицензии GPLv2.

Постановка задачи: разработать контентный фильтр ресурсов сети интернет, к которой подключено большинство средних школ в рамках проекта “Образование”.

Целью работы является защита школьников от информации, пропаганды и агитации, наносящих вред их здоровью, нравственному и духовному развитию.

Научная и практическая новизна работы заключается в использовании в контентном фильтре открытого программного обеспечения, которое является бесплатным и могут использоваться в любых целях на основе свободной лицензии GPLv2.

Практическая ценность работы заключается в доступности разработанной системы контентной фильтрации ресурсов сети интернет.

Результатом работы является система фильтрации сети интернет с использованием компьютерного оборудования на базе процессора Intel и методические рекомендации по использованию вариантов установки программного комплекса.

Программно-аппаратный комплекс контентной фильтрации сети интернет

Для создания контентного фильтра сети интернет целесообразно использовать свободное программное обеспечение на базе операционной системы GNU/Linux, а в качестве аппаратной платформы - компьютерное оборудование на базе процессора Intel, как самого распространённого на сегодняшний день. Также для разработки контентного фильтра потребуются начальные знания программирования для написания не сложных скриптов для командной оболочки UNIX.

1 Анализ вариантов развертывания программного комплекса

Возможны три варианта установки программного комплекса контентной фильтрации. Первый вариант - это установка фильтра на одном из компьютеров локальной сети. Данный вариант является самым простым и дешёвым. Нет необходимости выделять отдельный компьютер для данной задачи. Он может использоваться не только как фильтр, но и как рабочая станция. На компьютере достаточно будет установить только прокси-сервер и программу контентной фильтрации. Но производительность такого решения будет не высока, так как часть процессорного времени будет тратиться на решение учебных задач, не связанных с контентной фильтрацией. Второй вариант - выделение отдельного компьютера под данную задачу и подключение его в локальную сеть, также как и рабочие станции. В этом случае все вычислительные ресурсы компьютера будут использоваться только для фильтрации трафика. Недостатком указанных вариантов является необходимость настройки всех интернет-браузеров на работу через прокси-сервер, установленный на таком компьютере. В этом случае появляется возможность полностью обойти фильтр, указав в настройках браузера прямое соединение с интернетом. Третий вариант - это создание шлюза, который бы помимо функций маршрутизатора, dhcp-сервера, firewall и NAT для локальной сети, осуществлял контентную фильтрацию трафика. Преимуществом данного варианта является невозможность обхода фильтра со стороны локальной сети и полный контроль над сетевым трафиком. Недостатком является необходимость выделения под данную задачу отдельного компьютера.

При выборе сборки GNU/Linux целесообразно отдать предпочтение серверным редакциям или дистрибутивам, которые можно установить без графической оболочки, например Archlinux, Gentoo. Для управления сервером достаточно эмулятора консоли, отсутствие графического интерфейса позволит максимально использовать ресурсы компьютера для основной задачи. При ограниченности ресурсов выделенного компьютера последнее является существенным.

В качестве DHCP-сервера можно использовать Dnsmasq. Этот сервер с открытыми исходными кодами не требователен к ресурсам компьютера, не сложен в настройках и предназначен для работы в небольших сетях, подобных локальной сети. Dnsmasq поддерживает статические и динамические ip-адреса.

На шлюз в обязательном порядке устанавливается прокси-сервер Squid. Прокси-сервер необходим для работы интернет-фильтра. Также он осуществляет кэширование интернет-страниц, что позволяет экономить трафик и ускорять загрузку сайтов. Squid имеет встроенные возможности по блокировке сетевого трафика, что можно использовать при решении поставленных задач. Squid является самым распространённым прокси-сервером с

открытыми исходными кодами для GNU/Linux, обладает высокой производительностью, гибкими настройками и высокой надёжностью в работе.

В качестве интернет-фильтра лучше всего подходит Dansguardian. Это кроссплатформенная программа является бесплатной для некоммерческого использования. Она может, как и Squid, блокировать доступ к сайтам по именам домена или ip-адресам, но самым главным ее достоинством является наличие фильтрации контента по ключевым словам. Т. е. Dansguardian может сканировать содержимое страницы и заблокировать её показ пользователю, если она содержит слова, нарушающие определённые критерии. Данная программа является ключевой в построении контентного фильтра интернета. Dansguardian не скачивает страницы на прямую, а работает поверх прокси-сервера.

Для усиления возможностей интернет-фильтра дополнительно можно установить бесплатную серверную программу Vfilter [3]. Vfilter - это фильтрующий прокси-сервер, предназначенный в основном для блокировки рекламного кода на сайтах. Он может работать с чёрными списками, но главная и существенная его особенность - это наличие модуля эвристического анализа для выявления кода рекламы, отсутствующего в чёрных списках, и его блокировка. Это существенно улучшает работу фильтра.

Помимо готовых решений понадобятся программы в виде shell-скриптов по управлению настройками интернет-фильтра, программа обновления по расписанию черного списка сайтов для squid, программа создания белого списка сайтов для dansguardian и программа-редиректор на языке программирования perl.

2 Технология каскадирования серверов

Для осуществления взаимодействия всех серверов как единого программного комплекса применяется технология каскадирования серверов. Схема работы интернет-фильтра и интернет-шлюза, а также порядок прохождения запросов и взаимодействия всех программ представлена на рисунке 1.

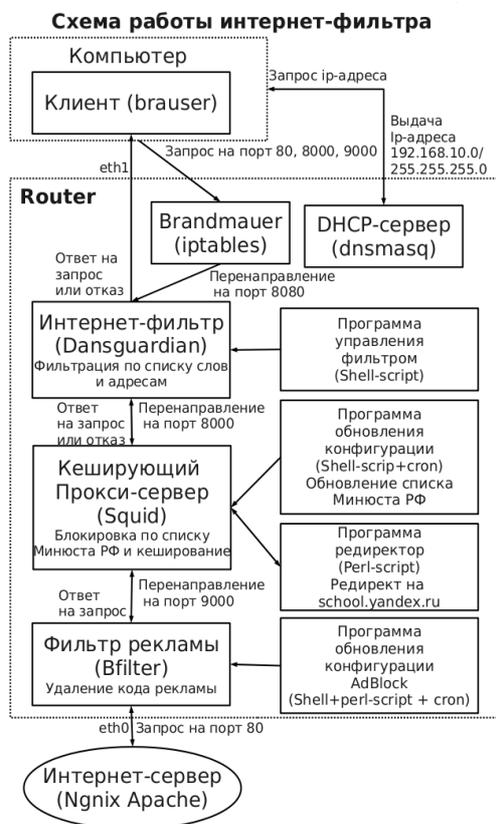


Рисунок 1 - Схема работы интернет- фильтра и интернет- шлюза

Для усиления защиты локальной сети школы от интернет-атаки, нежелательной электронной рассылки целесообразно установить дополнительно на интернет-шлюзе антивирусное программное обеспечение с открытыми исходными кодами clamav, а также систему обнаружения вторжений snort.

Заключение

На основе операционной системы GNU/Linux можно создать мощный программно-аппаратный комплекс на базе одного компьютера, который сможет решать сразу все задачи по организации локальной сети в школе, подключению к сети интернет, фильтрации трафика и защиты от внешних угроз. Решение проблемы фильтрации сетевого трафика сети интернет с использованием открытого программного обеспечения представляет собой выгодную и достойную альтернативу коммерческим программам.

Литература

1. Федеральный закон «Об основных гарантиях прав ребёнка в Российской Федерации» от 24 июля 1998 г. № 124-ФЗ.
2. В.Столлинг Свременные компьютерные сети.- Питер, 2003.- 783с.
3. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: учебное пособие.- Питер, 2010.- 943с.

ПОСТАНОВКА И ФОРМАЛИЗАЦИЯ ЗАДАЧИ ПРОЕКТИРОВАНИЯ СЕТИ ЭНЕРГОСНАБЖЕНИЯ

Кузьмина И.А.

Научный руководитель: д.ф-м.н., профессор, Карпенко А.П.

МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

STANDING AND FORMALIZATION OF THE PROBLEM OF POWER SUPPLY NETWORK DESIGN

Kuzmina I.A.

Supervisor: Dr., Prof., Karpenko A.P.

MSTU, Moscow, Russia

Аннотация

В статье рассматривается вопрос формализации задачи перспективного развития сети энергоснабжения. Дана математическая постановка задачи в виде многокритериальной задачи структурно-параметрической оптимизации, рассмотрены некоторые подходы к ее решению.

Abstract

The report focuses on formalization of the problem of perspective development of a power supply network. Its mathematical formulation is presented in the form of a multiobjective structural and parametric optimization problem, with several approaches to its solution being considered.

Введение

В общем случае задачу проектирования развития электросети можно рассматривать как задачу установления пути ее оптимального развития с точки зрения конфигурации, загрузки оборудования, параметров, нагрузок подстанций и т.д., а также определения необходимости и сроков ввода объектов нового строительства. Оптимальное развитие подразумевает обеспечение наилучших технико-экономических показателей. При этом сеть должна удовлетворять имеющимся требованиям к надежности и качеству энергоснабжения, воздействию на окружающую среду.

1 Модель сети электроснабжения

Сеть электроснабжения представляет собой совокупность объектов следующих типов:

- T – «трансформаторная подстанция» (ТП);
- R – «распределительная подстанция» (РП);
- C – «потребитель»;
- L – «кабельная линия» (КЛ).

Исходная сеть электроснабжения с подключенной нагрузкой представляет собой граф

$$G^{исх} = (R^{исх}, T^{исх}, L^{исх}),$$

где $R^{исх} = \{R_i^{исх} \in R, i \in [1..|R^{исх}|]\}$ – исходное множество узлов сети типа R ;

$T^{исх} = \{T_i^{исх} \in T, i \in [1..|T^{исх}|]\}$ – исходное множество узлов сети типа T ;

$L^{исх} = \{L_{i,j}^{исх} \in L, i \in [1..|R^{исх}|], j \in [1..|R^{исх}| + |T^{исх}| + |C^{под}|]\}$ – исходное

множество всех КЛ сети.

Элементы множества $\mathbf{T}_{исх}$ и $\mathbf{R}_{исх}$ являются вершинами графа $\mathbf{G}_{исх}$, элементы множества $\mathbf{LH}_{исх}$ соответствует его дугами.

Совокупность всех подключаемых к сети электроснабжения потребителей определяет множество $\mathbf{C}^{подкл} = \{C_i^{подкл} \in \mathbf{C}, i \in [1...|\mathbf{C}^{подкл}|]\}$.

Подключение новых потребителей к существующей сети электроснабжения требует ее модификации. Обозначим граф итоговой сети как

$$\mathbf{G}_{итог} = (\mathbf{R}_{итог}, \mathbf{T}_{итог}, \mathbf{L}_{итог}),$$

где $\mathbf{R}_{итог}$, $\mathbf{T}_{итог}$ – вершины графа, $\mathbf{L}_{итог}$ – дуги этого графа. Здесь $\mathbf{R}_{итог} = \{R_i^{итог} \in \mathbf{R}, i \in [1...|\mathbf{R}_{итог}|]\}$ – совокупность РП, полученная при преобразовании графа сети $\mathbf{G}_{исх} \rightarrow \mathbf{G}_{итог}$; $\mathbf{T}_{итог} = \{T_i^{итог} \in \mathbf{T}, i \in [1...|\mathbf{T}_{итог}|]\}$ – совокупность ТП, способных обеспечить питание всего множества потребителей, как ранее подключенных к сети электроснабжения, так и подключаемых потребителей множества $\mathbf{C}^{подкл}$;

$$\mathbf{L}_{итог} = \{L_{i,j}^{итог} \in \mathbf{L}, i \in [1...|\mathbf{R}_{итог}|], j \in [1...|\mathbf{T}_{итог}| + |\mathbf{C}^{подкл}|]\} -$$

множество всех КЛ, соединяющих объекты множеств $\mathbf{R}_{итог}$, $\mathbf{T}_{итог}$ и $\mathbf{C}^{подкл}$.

Фрагмент топологии итоговой сети электроснабжения, представлен на рис. 1

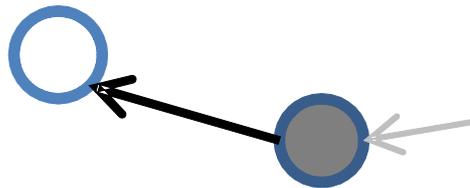


Рисунок 1 - Схема фрагмента сети электроснабжения с подключенными потребителями и построенными новыми ТП и РП

○ – потребитель; ● – РП; ◐ – ТП; — – КЛ; ◐ – новая ТП; ◑ – новая РП; — – новая КЛ

Вектор варьируемых параметров представим в виде

$$\mathbf{X} = (X_i, X_{1_j}, X_{2_j}, (x_k, y_k), X_{нов}^R, X_{нов}^T), \quad (1)$$

$$i \in [1...|\mathbf{C}^{подкл}|], j \in [1...|\mathbf{T}^{нов}|], k \in [1...|\mathbf{R}^{нов}| + |\mathbf{T}^{нов}|],$$

где X_i – номер ТП/РП, к которой будет подключен потребитель $C_i^{подкл}$; $X_{нов}^R$, $X_{нов}^T$ – количество новых ТП и РП; (x_k, y_k) – географические координаты строительства новых ТП/РП; X_{1_j}, X_{2_j} – номера ТП/РП, к которым будут подключены новые ТП подстанции.

Ограничения

На ряд параметров объектов типов \mathbf{R} , \mathbf{T} , \mathbf{L} сети электроснабжения наложены ограничения вида

$$W_i(\mathbf{X}) \geq 0. \quad (2)$$

Здесь $\mathbf{W}(\mathbf{X}) = (W_1(\mathbf{X}), W_2(\mathbf{X}), \dots, W_{|\mathbf{W}|}(\mathbf{X}))$, а неравенство (2) выполняется покомпонентно.

Требование (2) задает область допустимых значений вектора варьируемых параметров:

$$\mathbf{D}_X = \{ \mathbf{X} | \mathbf{W}_X(\mathbf{X}) \geq 0 \}.$$

Критерии оптимальности

Определены частные критерии оптимальности развития сети электроснабжения $\mathbf{Z}(\mathbf{X}) = (Z_1(\mathbf{X}), Z_2(\mathbf{X}), \dots, Z_{|\mathbf{Z}|}(\mathbf{X}))$.

Не нарушая общности положим, что на первые $|\mathbf{Z}|$ критериев наложены критериальные ограничения вида

$$Z_i^- \leq Z_i(\mathbf{X}) \leq Z_i^+, i \in [1 \dots |\mathbf{Z}|], \quad (3)$$

где Z_i^-, Z_i^+ – минимальное и максимальное значения i -ого критерия оптимальности соответственно, $|\mathbf{Z}|$ – число критериальных ограничений.

Критериальные ограничения (3) формируют еще одну область допустимых значений вектора варьируемых параметров $\mathbf{D}_Z = \{ \mathbf{X} \mid Z_i^- \leq Z_i(\mathbf{X}) \leq Z_i^+, i \in [1 \dots |\mathbf{Z}|] \}$.

Постановка задачи

Задачу проектирования сети энергоснабжения ставим в виде

$$\mathbf{Z}(\mathbf{X}^*) = \min_{\mathbf{X} \in \mathbf{D}} \mathbf{Z}(\mathbf{X}), \quad (4)$$

где \mathbf{X}^* – допустимое оптимальное значение вектора варьируемых параметров; \mathbf{D} – итоговое множество допустимых значений этого вектора, $\mathbf{D} = \mathbf{D}_X \cap \mathbf{D}_Z$.

С содержательной точкой зрения, задача (4) представляет собой задачу многокритериальной структурно-параметрической оптимизации.

2 Сведение задачи многокритериальной оптимизации к однокритериальной задаче

Критерии оптимальности $Z_1(\mathbf{X}), Z_2(\mathbf{X}), \dots, Z_{|\mathbf{Z}|}(\mathbf{X})$ в общем случае противоречат друг другу. Поэтому решение задачи (4) может быть только компромиссным [1].

В настоящее время существует большое число методов решения многокритериальных задач, а также различные подходы к их классификации. На верхнем уровне иерархии методы решения задач многокритериальной оптимизации классифицируем следующим образом [2].

1) Методы поиска решения без участия лица принимающего решения (ЛПР) [3].

2) Методы поиска решения с участием ЛПР. В зависимости от варианта вовлечения ЛПР строятся априорные, апостериорные или интерактивные методы многокритериальной оптимизации [2].

Кроме того, известные методы многокритериальной оптимизации можно разделить на две группы.

1) Методы многокритериальной оптимизации, которые не производят свертывание частных критериев в скалярный. К данным методам можно отнести метод справедливого компромисса, метод отклонения от идеальной точки, метод последовательных уступок, метод анализа иерархий и т.д. [4]. Данные методы позволяют получать допустимые решения, но требуют высокой квалификации ЛПР.

2) Методы, сводящие многокритериальную задачу к однокритериальной путем свертывания векторного критерия в скалярный. Наиболее часто используемыми методами являются метод главного критерия, аддитивная линейная свертка, мультипликативная свертка, свертка Гермеiera.

Результатом применения любой из сверток является сведение многокритериальной задачи (4) к однокритериальной детерминированной задаче глобальной оптимизации вида

$$Z(\mathbf{X}^*) = \min_{\mathbf{X} \in \mathbf{D}} Z(\mathbf{X}). \quad (5)$$

3 Сведение задачи перспективного развития сети энергоснабжения к задаче дискретного программирования

Значения компонентов вектора варьируемых параметров (3) X_i, X_{1j}, X_{2j} являются элементами множества уникальных номеров ТП и РП, которое представляет собой

совокупность дискретных величин. На переменные $X_{\text{нов}}^R$ и $X_{\text{нов}}^T$ наложено ограничение целочисленности. Координаты (x_k, y_k) выбираются из непрерывного диапазона значений. Таким образом, задача (5) представляет собой *однокритериальную задачу частично-дискретного программирования*.

Для решения задач частично-дискретного программирования используются три основных подхода [5]:

- решение исходной задачи без каких-либо упрощений;
- сведение к непрерывной задаче;
- сведение к задаче дискретного программирования.

Решение задачи (5) как задачи частично-дискретного программирования позволяет производить выбор значений координат (x_k, y_k) из непрерывного диапазона значений. Однако, учитывая прогнозный характер определения подключаемой к сети нагрузки, затраты на вычисление точных значений координат строительства новых объектов являются неоправданными.

Сведение частично-дискретной задачи к задаче непрерывного программирования приводит, во-первых, к увеличению вычислительной сложности задачи. Во-вторых, близость значений дискретных величин в пространстве поиска не соответствует в общем случае близости решений в пространстве критериев оптимальности. В результате данный подход может дать решение, сильно отличающееся от оптимального, или решение, выходящее за границы области допустимых значений \mathbf{D} . Таким образом, применение данного подхода к задаче (5) не целесообразно.

Представим области возможного строительства новых ТП и РП совокупностью прямоугольников $\mathbf{O} = \{O_i, i \in [1...|\mathbf{O}|]\}$. Сведение задачи (5) к задаче дискретного программирования произведем дискретизацией областей $O_i, i \in [1...|\mathbf{O}|]$. Дискретно-аппроксимированное множество \mathbf{O} , состоящее из множества точек – возможных мест строительства новых объектов, обозначим $\bar{\mathbf{O}} = \{(x_j, y_j), j \in [1...|\bar{\mathbf{O}}|]\}$.

В результате указанной дискретизации множество \mathbf{D}_X допустимых значений вектора варьируемых параметров преобразуется в множество $\bar{\mathbf{D}}_X$. Таким образом, задачу (5) сводим к *однокритериальной детерминированной дискретной задаче* условной оптимизации вида

$$Z(\mathbf{X}^*) = \min_{\mathbf{X} \in \bar{\mathbf{D}}} Z(\mathbf{X}),$$

$$\bar{\mathbf{D}} = \bar{\mathbf{D}}_X \cap \mathbf{D}_Z.$$

Заключение

Задача перспективного развития сети энергоснабжения в общем случае является сложной многопараметрической задачей. Предложенный подход представления задачи в виде однокритериальной дискретной детерминированной задачи позволяет получить эффективные методы ее решения.

Литература

1. Соловьев В.И. Методы оптимальных решений: Учебное пособие. – М.: Финансовый университет, 2012.
2. Лотов А.В., Поспелова И.И. Многокритериальные задачи принятия решений: Учебное пособие. – М.: МАКС Пресс, 2008.
3. Мухлисуллина Д.Т., Моор Д.А. Анализ эффективности различных сверток критериев оптимальности в задаче многокритериальной оптимизации. – М. Наука и образование 04, апрель 2010.
4. Никонов О.Я., Подоляка О.А., Подоляка А.Н., Скакалина Е.В. Математические методы решения многокритериальной задачи о назначениях. – [Вестник Харьковского национального автомобильно-дорожного университета](#), №5, 2011, С.103-112.
5. Корбут А.А., Финкельштейн Ю.Ю. Дискретные задачи математического программирования – М.: ВИНТИ, 1967. С. 59-110.

РАЗРАБОТКА ПРОГРАММНОЙ РЕАЛИЗАЦИИ КОМБИНИРОВАННОГО МЕТОДА ДЛЯ СОКРЫТИЯ ИНФОРМАЦИИ, ПРИМЕНЯЮЩЕГО ЦИФРОВУЮ ГОЛОГРАММУ ФУРЬЕ НА ОСНОВЕ ДИСКРЕТНОГО КОСИНУСНОГО ПРЕОБРАЗОВАНИЯ И АЛГОРИТМА НА ОСНОВЕ КВАНТОВАНИЯ.

Логунова А.О.

Научный руководитель: к.т.н., доцент, Волосатова Т.М.

МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

DEVELOPMENT PROGRAMME IMPLEMENTATION COMBINED METHOD FOR EMBEDDING INFORMATION, APPLYING THE DIGITAL HOLOGRAM OF FOURIER ON THE BASIS OF DISCRETE COSINE TRANSFORMATION AND ALGORITHM ON THE BASIS OF QUANTIZATION.

Logunova A. O.

Supervisor: Ph.D. assistant professor Volosatova T.M.

MSTU, Moscow, Russia

Аннотация

В статье рассматриваются алгоритмы стеганографии. Подробно исследован комбинированный метод сокрытия информации, применяющий цифровую голограмму Фурье на основе дискретного косинусного преобразования и алгоритм на основе квантования, представлена его программная реализация. В статье также приведены результаты работы созданного программного средства. На их основании проведено исследование возможности использования данного метода для защиты данных в САПР. В заключение представлены рекомендации по использованию данных алгоритмов.

Annotation

The algorithms of steganography are examined in the article. In it is in detail investigated the combined method of the concealment of information, applying the digital hologram of Fourier on the basis of discrete cosine transformation and algorithm on the basis of quantization. Its program realization is represented. Examples of software's output were also included into the article. The possibility of applying these techniques to data protection in CAD systems was analyzed on the basis of these examples. Recommendations on the use of the specified algorithms were given in the conclusion.

Введение

Целью работы является разработка приложения для сокрытия информации, реализующего синтез цифровой голограммы Фурье на основе дискретного косинусного преобразования и алгоритма на основе квантования. В основе комбинированного метода заложено два этапа – внедрение в контейнер защищаемых данных алгоритмом на основе квантования, а затем получение цифрового голографического изображения.

В рамках работы выполнены исследование методов стеганографии и их программная реализация. Разработано программное средство с графическим интерфейсом, позволяющее преобразовать изображение с помощью голографического метода, а также скрывать и извлекать информацию с помощью алгоритма на основе квантования. Результаты тестирования разработанного ПО представлены в данной статье.

1 Постановка задачи

Стеганография – это расширяемая совокупность методов и средств передачи (хранения) скрытой информации, при которых скрытый канал организуется на базе и внутри открытого канала с использованием особенностей восприятия информации.

При проведении анализа принято считать, что основными стеганографическими понятиями являются сообщение и контейнер.

Контейнером будем называть несекретные данные, используемые для сокрытия сообщений. Сообщением будем называть секретные данные, наличие которых в контейнере необходимо скрыть. Ключом будем называть секретную информацию, известную только законному пользователю.

В данной работе сообщением будет являться модифицированное изображение, полученное с помощью алгоритма на основе квантования. Ключом будут являться параметры, с помощью которых будет восстанавливаться изображение. Контейнером будет являться изображение, в которое будет встроено сообщение.

2 Алгоритм на основе квантования

Под квантованием понимается процесс сопоставления большого множества значений с некоторым конечным множеством чисел, при этом происходит уменьшение объема информации за счет ее искажения. Квантование находит применение в алгоритмах сжатия с потерями. Различают скалярное и векторное квантование. При векторном квантовании, в отличие от скалярного, происходит отображение не отдельно взятого отсчета, а их совокупности – вектора, кроме того векторное квантование эффективнее скалярного по степени сжатия (обладает большей сложностью).

В кодере квантователя вся область значений исходного множества делится на интервалы, и в каждом интервале выбирается число его представляющее. Это число есть кодовое слово квантователя и обычно бывает центроидом интервала квантования. Множество кодовых слов называется кодовой книгой. Все значения, попавшие в данный интервал, заменяются в кодере на соответствующее кодовое слово. В декодере принятому числу сопоставляется некоторое значение. Интервал квантования обычно называют шагом квантователя. Встраивание информации с применением квантования относится к нелинейным методам.

Передаваемое сообщение имеет ограниченную энергию для выполнения требования его незаметности. Помехами являются исходный сигнал и шум обработки. Кодеру исходный сигнал известен, декодер должен извлечь ЦВЗ без знания обеих составляющих помех. Существуют многочисленные способы, направленные на борьбу с помехами, заключающиеся в применении различных структурированных квантователей.

Наиболее предпочтительно внедрение информации в спектральную область изображения. Если при этом используются линейные методы, то встраивание ЦВЗ производят в средние полосы частот (энергия изображения сосредоточена, в основном, в низкочастотной (НЧ) области, значит, в детекторе ЦВЗ в этой области наблюдается сильный шум самого сигнала). В высокочастотных (ВЧ) областях большую величину имеет шум обработки, например, сжатия. В отличие от линейных, нелинейные схемы встраивания информации могут использовать НЧ области, так как мощность внедряемого ЦВЗ не зависит от амплитуды коэффициентов (для малой и большой амплитуды они обрабатываются одинаково).

В данной работе применялось встраивание на основе смешанного алгоритма скалярного и векторного квантования.

3 Голограмма Фурье

3.1 Цифровая Фурье - голография

В реализуемом методе, основанном на применении Фурье – голограммы заложены два этапа – внедрение в контейнер зашифрованных данных, а затем получение цифрового голографического изображения. Известно, что в самом общем случае формирование голограммы – это своеобразная модуляция голографируемого объекта.

Вне зависимости от того, в какой области, пространственной или частотной, выполняется модуляция изображения, в основе этих методов лежит интерференция между волновым полем, рассеянным ЦВЗ и пространственной несущей. Здесь проявляется известное свойство преобразования Фурье, которое характеризуется сдвигом спектра сигнала при умножении сигнала на гармоническое колебание. Используя преобразование Фурье объектов, имеющих сдвиг в частотной плоскости относительно начала координат, получаем интерференцию волнового поля, рассеянного объектом, с опорным волновым пучком, то есть комплексную голограмму. Процесс восстановления объекта сопровождается появлением действительного и мнимого изображения. Для моделирования голографических процессов применяется способ преобразования Фурье простых объектов, имеющих некоторое смещение относительно начала координат. Такие голограммы получили название цифровых Фурье - голограмм. Их можно интерпретировать следующим образом:

1. Транспаранту T_r соответствуют файлы с изображениями динамически меняющегося тест – объекта;
2. Объектив HN моделируется преобразованием Фурье от функции, кодирующей изображения;
3. Взаимодействие ОП с распределением поля, моделируемым преобразованием Фурье от функции, кодирующей изображения, описывается сложением с полем комплексной амплитуды наклонной под произвольно выбранным углом плоской волны.

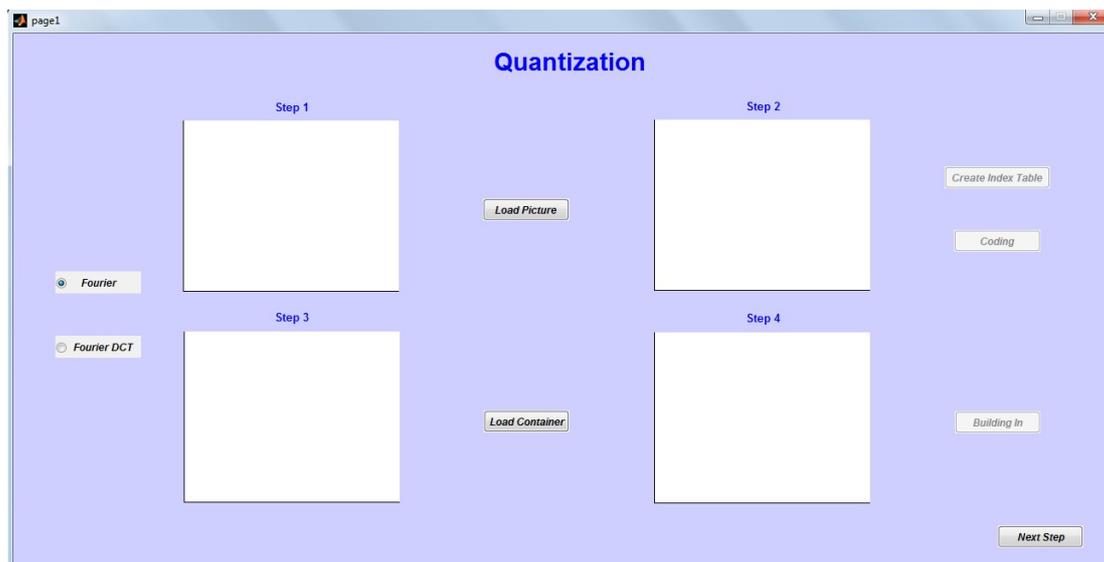
3.2 Синтез цифровой голограммы Фурье

Цифровые голограммы Фурье моделируют классические процессы и представляют собой запись пространственного преобразования Фурье от рассеянного виртуальным предметом светового поля со сдвинутой виртуальной пространственной несущей. В общем случае алгоритм синтеза изображения ЦВЗ для получения голограммы $h(x,y)$ выглядит следующим образом:

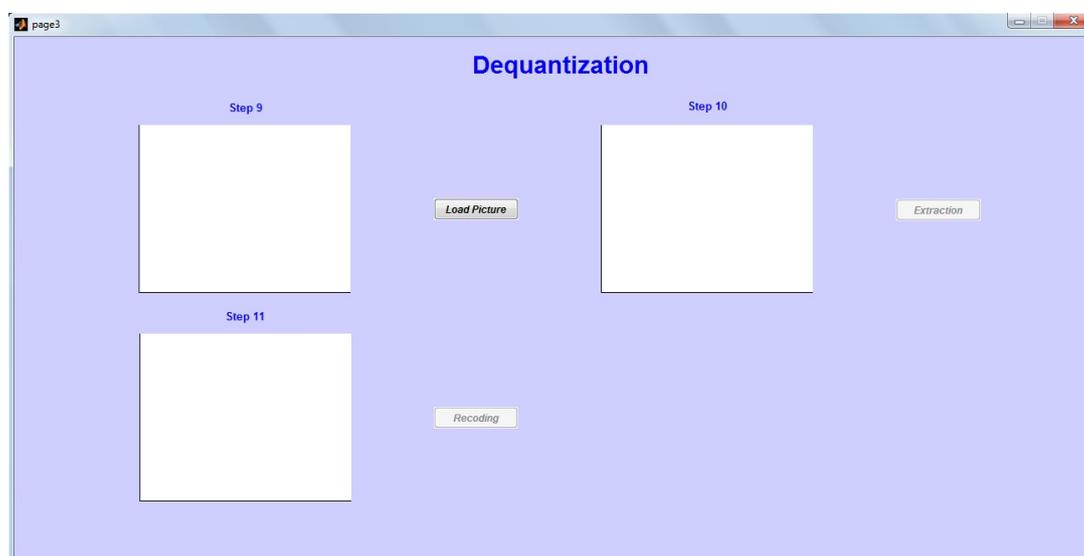
1. В виртуальной предметной области (x,y) формируется изображение ЦВЗ $W(x,y)$;
2. ЦВЗ переносится в плоскость пространственных частот (u,v) со смещением относительно осей координат на значения M, N : $W(u-M, v-N)$;
3. Создается зеркальное изображение ЦВЗ и получается сумма двух изображений: $\mathcal{F}\{h(x,y)\} = \tilde{W}(u,v) \neq W(u-M, v-N) \neq W(u+M, v+N)$, где γ – коэффициент усиления;
4. Для получения исходного распределения $h(x,y)$ выполняется обратное преобразование Фурье: $h(x,y) = \mathcal{F}^{-1}\{\tilde{W}(u,v)\}$, где \mathcal{F}^{-1} - оператор обратного преобразования Фурье.

4 Описание программной реализации

Программы реализованы в среде Matlab. На рис. 1 представлен графический интерфейс системы, позволяющей пользователю скрыть данные в изображение. Для встраивания данных пользователь должен выбрать изображение, в которое они будут встраиваться.



а) Интерфейс программы встраивания данных



б) Интерфейс программы извлечения данных

Рисунок 1 – Графический интерфейс программы, реализующей алгоритм на основе квантования

Графический интерфейс визуализации результатов позволяет пользователю преобразовывать изображение, т.е. создать голограмму Фурье.

5 Сравнение результатов

Из приведённых изображений видно, что алгоритм сокрытия данных на основе квантования даёт результат с погрешностью.



а) Встраиваемое изображение (Сообщение)



б) Контейнер



в) Контейнер со встроенным сообщением



г) Извлеченное сообщение

Рисунок 3 – Этапы встраивания и извлечения данных, используя алгоритм на основе квантования

Непосредственное использование этого метода в проектной документации САПР приводит к порче файлов и потере данных (изображение может быть восстановлено, но некоторые фрагменты будут искажены).



а) Исходное изображение



б) Модифицированное изображение (Голограмма Фурье)



в) Восстановленное изображение

Рисунок 4 – Этапы создания голограммы и восстановления изображения на её основе

На рисунке 4 представлены результаты работы программы реализации голограммы Фурье.

Заключение

В результате проведения данной работы были разработаны программы для преобразования и сокрытия информации на основе комбинированного метода стеганографии. Достоинством данной реализации является то, что если при передаче данных будет обнаружено наличие встроенного сообщения в изображение, понадобятся параметры, с помощью которых станет возможным восстановление изображения из голограммы. Это потребует дополнительных вычислительных и интеллектуальных ресурсов. К недостаткам можно отнести возможность удаления защищённой информации или её части из файла при передаче по сети интернет. А также непосредственное использование этого метода в проектной документации САПР приводит к порче файлов и потере данных (изображение может быть восстановлено, но некоторые фрагменты будут искажены). В рамках данного проекта поставленные задачи были выполнены.

Литература

1. Гонсалес Р., Вудс Р., Эддинс С. Цифровая обработка изображений в среде MatLab. М.: Техносфера, 2006. 621 с.
2. Волосатова Т.М., Денисов А.В., Чичварин Н.В. Комбинированные методы защиты данных в САПР // Ж. Информационные технологии. 2012. № 5. Приложение.
3. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. М.: Салон-пресс, 2009. 272 с.
4. Poon T-Ch. Optical Scanning Holography with MATLAB. Virginia: Virginia Tech, 2007. 153 p.
5. Бадриев И.Б., Бандеров В.В., Задворнов О.А. Разработка графического пользовательского интерфейса в среде MatLab. Казань: Казанский государственный университет, 2010. 113 с.

ОБЗОР И ИСПОЛЬЗОВАНИЕ СКРИПТОВЫХ ЯЗЫКОВ ДЛЯ ВСТРАИВАЕМЫХ СИСТЕМ

Малышев К.А.

Научный руководитель: ассистент Арабов Д.И.

МГТУ им. Н.Э.Баумана, кафедра ИУ4, Москва, Россия

REVIEW AND USAGE SCRIPTING LANGUAGES FOR EMBEDDED SYSTEMS

Malyshev K.A.

Supervisor: assistant Arabov D.I.

MSTU named after N.E. Bauman, IU4 department, Moscow, Russia

Аннотация

В статье рассматривается возможность применения скриптовых языков во встраиваемых системах. Приведен обзор скриптовых движков для микроконтроллеров. На основании обзора выделен движок Pawn, которому уделено особое внимание. В качестве примера рассмотрена реализация системы управления сервоприводом с использованием скриптового языка Pawn на микроконтроллерной платформе.

Abstract

This article discusses the possibility of usage scripting languages in embedded systems. An overview of scripting engines for microcontrollers is presented. Pawn engine is selected, based on review. It receives special attention in this article. As an example, the implementation of the servo control system for microcontroller platform using the Pawn scripting language is considered.

Введение

Скриптовый язык (англ. scripting language, в русской литературе принято название язык сценариев) — высокоуровневый язык программирования для написания сценариев — кратких описаний действий, выполняемых системой [1].

При разработке сложных систем у разработчика может возникнуть необходимость абстрагироваться от аппаратной части разрабатываемого устройства и сконцентрироваться на алгоритмах и логике работы программы. Использование скриптового языка в таком случае позволяет решить эти проблемы. Реализовав все функции, которые зависят от конкретного аппаратного обеспечения на низкоуровневом языке, разработчик пишет всю логику работы устройства на скриптовом языке. Обычно в таком случае программы являются менее громоздкими и более легкими в восприятии, что упрощает отладку и сопровождение программного кода устройства. Конечно же, использование скриптового языка при разработке программ для встраиваемых систем имеет и недостатки. Основной недостаток при таком подходе – это меньшая скорость работы по сравнению с программами, написанными на более низкоуровневом языке.

Еще одна причина, по которой может понадобиться применение скриптового языка, – это ситуация когда разработчик не может предоставить заказчику все исходные коды проекта, но при этом ему крайне важно заложить возможность изменения алгоритмов поведения каких-либо систем. В этом случае основную логику работы можно переложить на открытый интерпретируемый код, а остальную часть реализовать на более низкоуровневом языке, без необходимости раскрывать исходный код [2].

1 Обзор скриптовых языков для микроконтроллеров

На сегодняшний день существует достаточно большое количество реализаций скриптовых языков и интерпретаторов для них. Среди языков выделяются как широко распространенные языки программирования вроде Basic, Forth, Python, Java, JavaScript, так и нестандартные языки программирования, которые похожи на язык C.

Рассмотрим некоторые реализации интерпретаторов. При рассмотрении реализаций интерпретаторов необходимо обращать внимание на следующие моменты:

- интерпретируемый язык;
- требования к размеру постоянной запоминающей памяти и оперативной памяти микроконтроллера;
- микроконтроллеры, на которые портирован данный интерпретатор;
- лицензия, по которой рассматриваемый интерпретатор распространяется;
- последнее обновление релиза.

Итак, первый рассматриваемый интерпретатор – это **eLua** – интерпретируемый язык программирования Lua для встраиваемых систем (от eLua – embedded Lua, embedded – встраиваемый) [3]. Достаточно известный скриптовый язык программирования Lua был портирован на микроконтроллеры, в результате чего появился проект eLua. Данный проект включает реализации для относительно большого числа микроконтроллеров и имеет хорошее количество дополнительных библиотек, которые позволяют обращаться к периферии микроконтроллера непосредственно из скрипта. Также этот интерпретатор позволяет экспортировать функции, написанные на более низком языке программирования, в скрипт, что позволяет уменьшать время выполнения программы в критических местах. Требования к памяти у eLua достаточно высокие: 128KB ROM и 32KB RAM – минимальные требования для запуска интерпретатора. Данный проект постоянно развивается, о чем свидетельствует дата последнего обновления – 27.05.13. Распространяется eLua по лицензии MIT License (свободная лицензия, пригодная для коммерческого пользования).

Существует еще одна реализация Lua для микроконтроллеров – это **uLua** [4]. По сравнению с eLua имеет ограниченный функционал, но при этом требует намного меньше памяти: 1KB RAM. Изначально писался под микроконтроллеры AVR, поэтому кроме этих микроконтроллеров и микроконтроллеров серии x86 в списке поддерживаемых на данный момент устройств больше ничего нет. Лицензия – MIT License. Дата последнего обновления – 25.02.13.

PyMite – интерпретатор широко распространенного скриптового языка Python [5]. Реализация данного интерпретатора портирована на большое количество микроконтроллеров. PyMite поддерживает большую часть конструкций и типов данных языка Python. Поддерживает интерактивную командную строку посредством интерфейса UART. Для использования аппаратнозависимых функций PyMite поддерживает экспорт функций, написанных на языке C. При всем этом интерпретатор требует 55KB ROM и 8KB RAM, что в несколько раз меньше требований eLua. Последнее обновление было совершено 26.12.10. Лицензия – GNU GPL v2 (свободное программное обеспечение), в отличие от MIT License не позволяет изменять лицензию кода.

Собственный язык для встраиваемых систем и абстрактную машину имеет проект **Pawn** [6]. Скриптовый язык напоминает язык C. В скрипты, как и в проектах рассмотренных выше, можно экспортировать функции, написанные на языке более низкого уровня. Одним из основных преимуществ Pawn является наличие механизма, который во время выполнения скрипта подгружает в ОЗУ только ту его часть, которая выполняется в данный момент. Этот механизм позволяет выполнять большие скрипты на системах с малым количеством ОЗУ. Так же исходный код скрипта перед выполнением компилируется в бинарный файл, что дает высокую скорость выполнения программы. В то же время абстрактная машина Pawn не очень требовательна к ресурсам: 48KB ROM и 10KB RAM. Последнее обновление было

зафиксировано 22.06.2012. Лицензия, по которой распространяется Pawn – Apache License 2.0 – лицензия аналогичная MIT License, позволяет изменять лицензию производных работ.

Espruino – проект, который включает в себя отладочную плату с установленным на нее микроконтроллером STM32F103RCT6, прошивку для этого микроконтроллера с интерпретатором языка JavaScript и среду разработки Espruino IDE [7]. Среда разработки дает возможность реализовывать алгоритм работы устройства в графическом режиме в виде блоков. Данный проект не позволяет экспортировать собственные функции, написанные на языке C и представляет собой только интерпретатор скриптов, написанных на JavaScript. Проект ориентирован на микроконтроллеры серии STM32Fxxx и Raspberry Pi. Минимальный размер требуемой памяти для STM32 не превышает 100KB ROM и 4KB RAM. Дата последнего обновления на момент написания статьи 31.01.14. Проект относительно новый и поэтому продолжает постоянно развиваться. Лицензия Mozilla Public License Version 2.0 – свободно распространяемый проект с возможностью объединяться с кодом под другой лицензией.

Виртуальная машина **Nano VM** была разработана для роботизированных платформ Asuro, c't-Bot и Nibo, использующий в качестве управляющей микросхемы микроконтроллер AVR, поэтому имеет низкие требования к памяти (8KB ROM и 256B RAM) [8]. Язык, используемый для программирования перечисленных выше платформ является частичной реализацией языка Java. Последние изменения произведены 13.06.06. Лицензия GPL License.

Embed VM – использует C-подобный синтаксис. Написана для AVR (требует всего 3KB ROM) [9]. Последнее изменение – 02.05.11. Лицензия ISC license.

ARMPIT SCHEME – интерпретатор языка Scheme (диалект языка Lisp) версии R5RS [10]. Портингован на большое количество микроконтроллеров и других встраиваемых систем. Требует сравнительно мало памяти: 32KB ROM и 8KB RAM. Последние изменения сделаны в феврале 2013 года. Распространяется по лицензии MIT License.

MMBasic – интерпретатор языка Basic [11]. Разрабатывался для аппаратной платформы Maximite на основе микроконтроллера серии Microchip PIC32 [12]. Занимает 94KB ROM и 9KB RAM. Последняя версия от 20.07.13. Распространяется свободно по собственной лицензии.

DK-BASIC – интерпретатор языка BASIC для 8-битных микроконтроллеров AVR [13]. По функциональности уступает MMBasic, но занимает 32KB ROM и меньше 1KB RAM. Последнее обновление относится к ноябрю 2009 года и по настоящее время находится в стадии альфа-версии. Лицензия – New BSD License.

Проект **AmForth** – интерпретатор достаточно известного скриптового языка Forth [14]. AmForth не позволяет встраивать интерпретатор в другой код, так же как и использовать в скрипте функции, написанные на другом языке, например C. Разработан AmForth для микроконтроллеров AVR ATmega и не предназначен для портирования на другие платформы. Интерпретатор требует 8KB ROM, 200B RAM и 80B EEPROM. Последнее обновление совершено 23.12.13, что на момент написания статьи относительно недавно. Лицензия GNU Public License v2.

Java Embedded – версия Java машины для встраиваемых систем [15]. В данный момент портирована на одноплатные компьютеры и микроконтроллеры STM32F207 и STM32F407. Имеет поддержку всей периферии выше названных микроконтроллеров и поддерживает большой набор функций стандартной версии Java – Java SE. Функциональность Java Embedded накладывает сравнительно жесткие ограничения на объем требуемой памяти. Для данной версии требуются микроконтроллеры с памятью не меньше 768KB. На текущий момент Java Embedded продолжает развиваться и портироваться на новые системы. Дата последнего обновления – январь 2013 года.

В таблице 1 представлены основные характеристики рассмотренных виртуальных машин.

Таблица 1 – Основные характеристики виртуальных машин для микроконтроллеров

Название	Куда портирована	Требования к памяти		Последнее обновление	Лицензия	Особенности
		ROM	RAM			
eLua	LM3S, AVR32, AT91SAM7X, LPC17xx, LPC24xx, LPC288x, STR7, STR9, STM32, i386	128KB	32KB	27.05.13	MIT License	Скриптовый язык Lua. Возможность экспорта C-кода.
uLua	x86, AVR		1KB	25.02.13	MIT License	Скриптовый язык Lua.
PyMite	семейство POSIX, WINDOWS, Arduino, AT91SAM7X, AVR, PIC24, STM32, LPC2368, LPC1768, Teensy, Mbed, Mosync	55KB	8KB	26.12.10	GNU GPL v2	Скриптовый язык Python. Возможность экспорта C-кода.
Pawn	ATmega128, LPC2138, LPC2106, MSP430	48KB	10KB	22.06.2012	Apache License 2.0	Собственный C-подобный язык. Механизм загрузки и выполнения скрипта по частям. Высокая скорость вып. байт-кода
Espruino	STM32F100 STM32F303 STM32F407 STM32F103 Raspberry Pi	100KB	4KB	31.01.14	Mozilla Public License Version 2.0	Скриптовый язык JavaScript. Собственная Espruino IDE с возможностью виз. программирования
Nano VM	AVR	8KB	256B	13.06.06	GPL License	Частичная реализация языка Java. Предназначена для роботизированных платформ Asuro, c't-Bot и Nibo.
Embed VM	AVR	3KB		02.05.11	ISC license	Собственный C-подобный язык
ARMPIT SCHEME	LPC2xxx, LPC1xxx, LPC4xxx, AT91SAMx, STM32Fxxx, LM3S19xx, LM3S6965, LM3S9B92, LM4F120, LM4F232, STR711, STR911, EP9302, S3C2410, OMAP3530, DM3730	32KB	8KB	xx.02.13	MIT License	Используется язык Scheme(R5RS)-диалект Lisp

MMBasic	PIC32	94KB	9KB	20.07.13	Свободно распространяемый (собственная лицензия)	Язык - BASIC
DK-BASIC	AVR	32KB	1KB	xx.11.09	New BSD License	Язык - BASIC
AmForth	AVR	8KB + 80B EEPROM	200B	23.12.13	GNU Public License v2	Язык - Forth
Java Embedded	STM32F207, STM32F407	≈700KB		xx.01.13		Виртуальная Java машина для STM

При выборе скриптового движка для микроконтроллера будем исходить из следующих требований:

- движок должен либо легко портироваться, либо поддерживать максимальное количество микроконтроллеров;
- движок должен требовать как можно меньше ресурсов и иметь как можно большую производительность;
- движок должен быть встраиваемым в основную программу;
- движок должен поддерживать возможность экспорта функций, написанных на языке более низкого уровня.

Всем требованиям одновременно лучше всего отвечает движок Pawn, пример работы с которым будет рассмотрен далее. В то же время еще одним его преимуществом является C-подобный синтаксис. Если же не требуется накладывать жесткие требования на память, то язык eLua тоже отвечает всем требованиям и при этом имеет большой набор встроенных функций.

2 Использование скриптового движка Pawn для реализации системы управления сервоприводом

Для того чтобы реализовать систему управления сервоприводом, необходимо решить следующие задачи:

- выбрать аппаратную платформу и собрать испытательный стенд;
- импортировать движок Pawn в проект для микроконтроллера, на котором основан испытательный стенд, указав необходимые параметры для препроцессора;
- реализовать аппаратнозависимые функции управления сервоприводом;
- реализовать алгоритм управления сервоприводом на языке Pawn.

В качестве аппаратной платформы выбрана отладочная плата Open207V на базе микроконтроллера STM32F207VC. Микроконтроллеры серии STM32 имеют хорошее соотношение цена/производительность и в настоящее время широко распространены и применяются во многих устройствах. Структура испытательного стенда приведена на рисунке 1.



Рисунок 1 – Структура испытательного стенда для системы управления сервоприводом

Для программирования и отладки программы для микроконтроллера используется JTAG-отладчик. Фотография испытательного стенда представлена на рисунке 2.

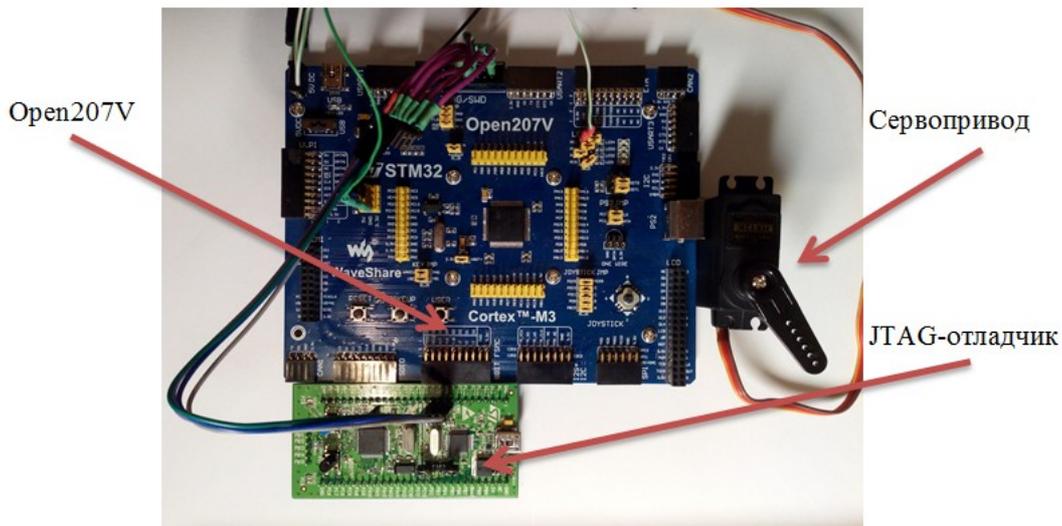


Рисунок 2 – Фотография испытательного стенда

Для импорта виртуальной машины Rawp в проект необходимо добавить следующие файлы:

- непосредственно саму виртуальную машину: amx.c и amx.h;
- определения переменных, зависящих от аппаратной части: osdefs.h;
- ассемблерный файл с реализацией функции выполнения команды виртуальной машины: amxhex_thumb2.s;
- файл, в который вынесены все параметры для препроцессора: rawnconf.h;
- реализацию аппаратнозависимых функций: drive.c и drive.h;
- файл со скриптом script.h.

Для оптимизации места, занимаемого виртуальной машиной, необходимо указать модули, которые требуется подключить в данном проекте. В данном проекте эта информация указана в файле rawnconf.h, содержимое которого приведено в листинге 1.

Листинг 1 – Содержимое файла rawnconf.h

```
#ifndef PAWNCONF_H_INCLUDED
#define PAWNCONF_H_INCLUDED

#define AMX_ANSIONLY //Использовать только ANSI, не использовать UNICODE
#define AMX_NO_PACKED_OPC //Виртуальная машина с прямым потоком команд
#define AMX_ASM //Использовать ассемблерную версию функции
выполнения команды виртуальной машины
#define AMX_NO_OVERLAY //Выключен механизм загрузки и выполнения кода по частям
#define AMX_NO_MACRO_INSTR //Не использовать макроинструкции
#define AMX_NODYNALOAD //Не использовать динамически подгружаемые
библиотеки
#define THUMB2 //Набор ассемблерных инструкций - THUMB2

#define AMX_CLEANUP //подключить функцию amx_Cleanup
#define AMX_DEFCALLBACK //функция amx_Callback по-умолчанию
#define AMX_EXEC //подключить функцию amx_Exec
#define AMX_INIT //подключить функции amx_Init и amx_InitJIT
#define AMX_REGISTER //подключить функцию amx_Register

#endif // PAWNCONF_H_INCLUDED
```

Приведенный набор параметров является минимальным для работы виртуальной машины и поддержки механизма экспорта функции, написанных на языке более низкого уровня.

Для управления сервоприводами реализовано несколько функций на языке C. Эти функции находятся в файле device.c и представлены в листинге 2.

Листинг 2 – Функции управления сервоприводами и функция их добавления в список функций виртуальной машины

```
#define JOY_A      GPIO_Pin_15
#define JOY_D      GPIO_Pin_13
//Функция получения статуса кнопки: 1 - нажата, 2 - отжата
static cell n_getButton(AMX *amx, const cell *params)
{
    /* getButton(button);
    * params[1] = button
    * D=1, A=2 */
    cell button = params[1];
    if(button==1)
        return !GPIO_ReadInputDataBit(GPIOD,JOY_D);
    else if(button==2)
        return !GPIO_ReadInputDataBit(GPIOB,JOY_A);
    else
        return 0;
}
//Функция управления сервоприводом: реализовано выставление положения ротора
//сервопривода в одно из двух положений
static cell n_setPos(AMX *amx, const cell *params)
{
    /* setPos(position);
    * params[1] = position */
    cell position = params[1];
    uint8_t i;
    if(position==1)
    {
        for(i=0;i<10;i++)
        {
            ms_delay(100);
            GPIO_SetBits(GPIOC,GPIO_Pin_6);
            us_delay(1000);
            GPIO_ResetBits(GPIOC,GPIO_Pin_6);
        }
    }
    if(position==2)
    {
        for(i=0;i<10;i++)
        {
            ms_delay(100);
            GPIO_SetBits(GPIOC,GPIO_Pin_6);
            us_delay(1500);
            GPIO_ResetBits(GPIOC,GPIO_Pin_6);
        }
    }
    return 0;}
//Функция, добавляющая функции выше в список функций, доступных для
//виртуальной машины, то есть их можно применять в самом скрипте
int amx_DriveInit(AMX *amx)
{
    static AMX_NATIVE_INFO drive_Natives[] = {
        { "getButton",  n_getButton },
        { "setPos",     n_setPos },
        { 0, 0 } /* terminator */
    };
    return amx_Register(amx, drive_Natives, -1);}

```

Все, функции, которые должны быть добавлены в список доступных для виртуальной машины, должны обязательно иметь заголовок как у функций `n_setPos` и `n_getButton`, которые представлены выше. Скрипт управления сервоприводом для виртуальной машины представлен в листинге 3.

Листинг 3 – Скрипт управления сервоприводом на языке Pawn

```
main()
{
    new D=1;    //код кнопки D
    new A=2;    //код кнопки A
    while(1)
    {
        if(getButton(D))
            setPos(1);
        if(getButton(A))
            setPos(2);}
}
```

Виртуальная машина выполняет байт-код, поэтому скрипт, представленный в листинге 3, был скомпилирован и затем записан в постоянную память микроконтроллера в виде массива байт.

Результаты тестирования на испытательном стенде показали, что система работает корректно: при нажатии кнопки D сервопривод поворачивает ротор в положение 1, а при нажатии кнопки A – в положение 2.

Литература

1. Сценарный язык – Электронный ресурс. Режим доступа: http://ru.wikipedia.org/wiki/Сценарный_язык – Проверено 03.02.2014.
2. Использование скриптового языка Embedded Lua во встраиваемых системах – Электронный ресурс. Режим доступа: <http://www.promwad.com/sites/default/files/publications/article-embedded-lua-script-ru.pdf> – Проверено 03.02.2014.
3. eLua – eluaproject – Электронный ресурс. Режим доступа: <http://www.eluaproject.net/> – Проверено 03.02.2014.
4. ulua – Google project hosting – Электронный ресурс. Режим доступа: <https://code.google.com/p/ulua/> – Проверено 03.02.2014.
5. python-on-a-chip – Google project hosting – Электронный ресурс. Режим доступа: <https://code.google.com/p/python-on-a-chip/> – Проверено 03.02.2014.
6. The Pawn language – Электронный ресурс. Режим доступа: <http://www.compuphase.com/pawn/pawn.htm> – Проверено 03.02.2014.
7. Espruino – Электронный ресурс. Режим доступа: <http://www.espruino.com/> – Проверено 03.02.2014.
8. NanoVM – Электронный ресурс. Режим доступа: <http://www.harbaum.org/till/nanovm/index.shtml> – Проверено 03.02.2014.
9. EmbedVM – Embedded Virtual Machine – Электронный ресурс. Режим доступа: <http://www.clsifford.at/embedvm/> – Проверено 03.02.2014.
10. A Scheme interpreters for ARM microcontroller – Электронный ресурс. Режим доступа: <http://armpit.sourceforge.net/> – Проверено 03.02.2014.
11. MMBasic – BASIC interpreter – Электронный ресурс. Режим доступа: <http://mmbasic.com/> – Проверено 03.02.2014.
12. The Colour Maximite – Электронный ресурс. Режим доступа: <http://geoffg.net/maximite.html> – Проверено 03.02.2014.
13. dk-basic – Google project hosting – Электронный ресурс. Режим доступа: <https://code.google.com/p/dk-basic/> – Проверено 03.02.2014.
14. AmForth – Электронный ресурс. Режим доступа: <http://amforth.sourceforge.net/> – Проверено 03.02.2014.
15. JAVA Embedded – Электронный ресурс. Режим доступа: <http://www.oracle.com/technetwork/java/embedded/overview/getstarted/index.html> – Проверено 03.02.2014.

ПРОГРАММНЫЙ МОДУЛЬ СИНТЕЗА ЦИФРОВЫХ ГОЛОГРАММ

Марченков А.М.

Научный руководитель к.т.н., доцент Волосатова Т.М.

МГТУ им. Н.Э. Баумана, Москва, Россия

SOFTWARE MODULE OF DIGITAL HOLOGRAM SYNTHESIS

Marchenkov A.M.

The scientific chief: Ph.D., associate professor Volosatova T.M.

Department RK6 BMSTU, Moscow, Russia

Аннотация

Описан метод обнаружения и распознавания подвижных объектов на основе формирования цифровой голограммы. Приведен алгоритм синтеза изображения для получения голограммы. Дано описание программного модуля синтеза цифровых голограмм.

Abstract

A method for detection and recognition of moving objects based on the formation of the digital hologram is disclosed. An image synthesis algorithm for hologram is described. Software module synthesis of digital holograms is described.

Введение

В настоящее время одним из направлений развития в области компьютерного зрения является модификация существующих алгоритмов для уменьшения вычислительных затрат, например, анализ возможности встраивания алгоритмов слежения непосредственно в камеру. Традиционные алгоритмы имеют высокую вычислительную сложность, а ресурсы однокристалльной платформы камеры ограничены. Таким образом, необходима разработка принципиально новых алгоритмов с большей вычислительной эффективностью [1].

Для определения перемещения объекта в пределах сцены обычно анализируется не вся сцена, а только особые точки. Особая точка сцены или точечная особенность – это такая точка сцены, изображение которой можно отличить от изображений всех соседних с ней точек.

Большинство современных алгоритмов слежения за особенностями опираются на работы Лукаса и Канаде. В трекаре Лукас-Канаде особенность считается только смещающейся, без искажений. Метод Лукас-Канаде часто применяется для поиска величины освещенности во всем изображении. Развитием этой работы являются следующие вариации трекаров: Томасе-Канаде (модификация метода Лукас-Канаде), Ши-Томасе-Канаде (метод учитывает аффинные искажения), Джин-Фаворе-Соатто (учитывает аффинные изменения освещенности).

Усложнение задачи слежения ведет к увеличению размерности системы линейных уравнений, к решению которой сводится данная задача, что приводит к увеличению вычислительной сложности. Подробнее задачи поиска точечных особенностей и варианты трекаров рассмотрены в работе [2]. Проведем исследования возможности уменьшения сложности задачи слежения с помощью применения голографических методов.

1 Метод обнаружения и распознавания подвижных объектов на основе формирования цифровой голограммы

Для получения цифровых голограмм часто используется подход, в котором к световой волне, рассеянной объектом, добавляется опорная волна. Цифровая голограмма Фурье является записью пространственного преобразования Фурье светового поля, рассеянного предметом, со сдвинутой несущей. Для объекта $W(u, v)$ цифровую голограмму, в общем

случае, можно представить как квадрат модуля интерференции комплексной функции $w(x, y) = |w(x, y)| \exp(j\Phi(x, y))$, где $\Phi(x, y)$ – аргумент комплексной функции $w(x, y)$ с когерентной опорной волной $\exp(j(Mx + Ny))$, где M и N – углы падения волны по нормали к плоскости регистрации голограммы. Распределение интенсивности в плоскости регистрации можно представить следующим образом:

$$h(x, y) = [|w(x, y)| \exp(j\Phi(x, y)) + \exp(j(Mx + Ny))]^2 - |w(x, y)|^2 + |w(x, y)| \cos(Mx + Ny + \Phi(x, y)) + 1.$$

Первый член в этом выражении характеризует дополнительную засветку голограммы пучком света от объекта. Он не содержит фазовой характеристики и не несет никакой информации о восстановленном изображении, поэтому его часто опускают. Следовательно, получаем:

$$h(x, y) = A_0 + |w(x, y)| \cos(Mx + Ny + \Phi(x, y)),$$

где постоянная составляющая A_0 – максимальное значение при $w(x, y) = w(0, 0)$. Таким образом, функция $h(x, y)$ является вещественной, положительной и содержит полную информацию (как амплитудную, так и фазовую) об изображении объекта $W(u, v)$. Преобразование Фурье от полученной функции голограммы $h(x, y)$ будет состоять из суммы двух изображений объекта $W(u, v)$, смещенных относительно начала осей координат на величины несущих частот M и N :

$$\mathfrak{F}\{h(x, y)\} = W(u, v) = W(u - M, v - N) + W(-u - M, -v - N) + A(u, v),$$

где \mathfrak{F} – оператор Фурье-преобразования, $A(u, v)$ – автокорреляционная функция. Автокорреляция $A(u, v)$ располагается в начале координат и подавляется как в цифровой голографии, так и в оптической. Второе изображение $W(-u-M, -v-N)$ является зеркальным отображением $W(u-M, v-N)$ относительно центра осей координат [3].

2 Алгоритм синтеза изображения для получения голограммы $h(x, y)$

Объект $W(x, y)$ расположен в предметной области (x, y) . После переноса его Фурье-образа в плоскость пространственных частот (u, v) со смещением относительно осей координат на значения M и N , объект примет вид $W(u - M, v - N)$.

После этого создается зеркальное изображение и вычисляется сумма двух изображений: $\mathfrak{F}\{h(x, y)\} = W(u, v) = \gamma W(u - M, v - N) + \gamma W(-u - M, -v - N)$, где γ – коэффициент усиления.

Чтобы получить исходное распределение достаточно выполнить обратное преобразование Фурье. При преобразовании изображения объекта следует учесть следующий фактор. Если предельные размеры изображения превысят размеры квадранта частотной плоскости, то произойдет перекрывание сигналов $W(u - M, v - N)$ и $W(-u - M, -v - N)$.

Для избежания избыточности представления $W(u, v)$ можно поступить следующим образом.

Представим комплексную функцию $h(x, y)$ в виде суммы вещественной и мнимой частей:

$$\text{Так } h(x, y) = h_c(x, y) + jh_s(x, y) = \frac{1}{2}|w(x, y)| \cos[xM + yN] + \frac{j}{2}|w(x, y)| \sin[xM + yN] + A_0 \quad \text{как}$$

спектральные сигналы $W(u, v)$ должны быть узкополосными, первые два члена этого

выражения связаны между собой преобразованием Гильберта (разложение на амплитудную и фазовую составляющие). Для восстановления $W(u, v)$ из голограммы будет достаточно выполнить преобразование Фурье только вещественной (косинусной), или только мнимой (синусной) части комплексной голограммы $h(x, y)$.

3 Описание программного модуля синтеза цифровых голограмм

На рисунке 1 приведен алгоритм синтеза цифровой голограммы. Стоит отметить, что при реализации синтеза голограммы на ЭВМ, увеличение длины и ширины тестового изображения необходимо для избежания наложения образов, которое упоминается выше. Изображение голограммы приведено на рисунках 2 и 3. Исходное и восстановленное изображения показаны на рисунке 4. Из рисунка видно, что восстановленное изображение соответствует оригиналу с небольшими искажениями.



Рисунок 1 - Алгоритм синтеза цифровой Фурье-голограммы

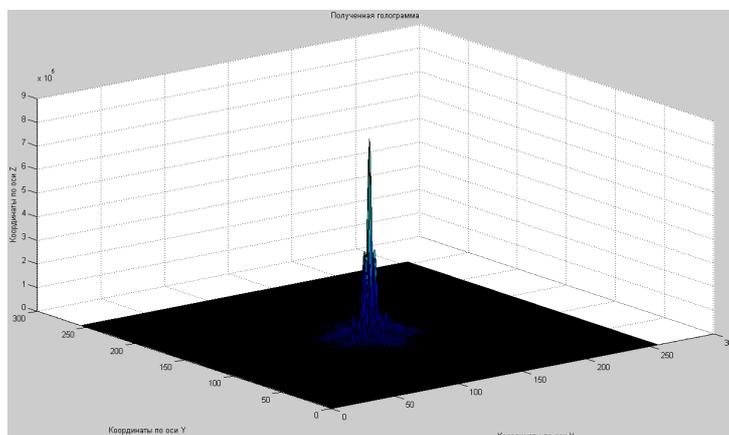


Рисунок 2 - Полученная голограмма

В работах [4, 5] проведен анализ голографических методов для случая переменного освещения, а также описан метод обнаружения и распознавания подвижных объектов. Принимая во внимание эти исследования, а так же то, что ресурсоемкость голографических методов меньше, чем у традиционных, можно сделать вывод, о возможности разработки алгоритма слежения с большей вычислительной эффективностью, чем у традиционных методов.

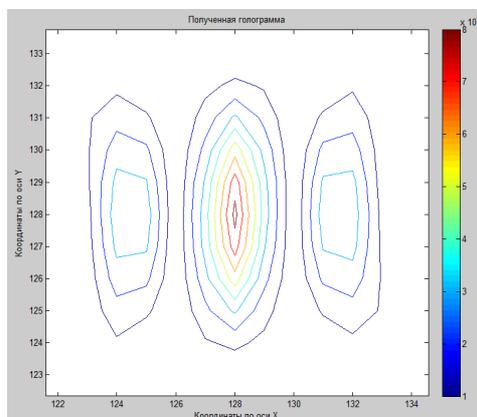


Рисунок 3 - Линии равного уровня полученной голограммы



Рисунок 4 - Объект и восстановленное изображение

Таким образом, для решения задач обнаружения и распознавания целесообразно применять голографический метод обнаружения и распознавания. Именно он позволяют обнаружить и распознавать движущиеся объекты в мгновенном поле зрения системы технического зрения, что обеспечивает его автосопровождение.

Литература

1. Птицын Н. Встроенная видеоаналитика для детектирования и сопровождения объектов при помощи многомасштабных признаков [Электронный ресурс] / Птицын Н. – Режим доступа: <http://synesis.ru/blog/article/vstroennaya-videoanalitika-dlya-detektirovaniya-i-soprovozhdeniya-obektov-pri-pomoshhi-mnogomasshtabnyix-priznakov> (22.12.2013)
2. Конушин. А. Слежение за точечными особенностями сцены [Электронный ресурс] // Компьютерная графика и мультимедиа. –2003 - №1 - Режим доступа: <http://cgm.computergraphics.ru/content/view/54>(22.12.2013)
3. Волосатова Т. М., Денисов А.В., Чичварин Н.В. Комбинированные методы защиты данных в САПР // «Информационные технологии» 2012. №5. (Приложение к журналу).
4. Марченков А. М. Модификация алгоритмов компьютерного зрения анализа динамических объектов для случая переменного освещения // «Научные технологии и интеллектуальные системы 2013»: сборник трудов. Москва, 2013. С. 105-108.
5. Волосатова Т. М., Марченков А.М., Чичварин Н.В. Исследования и разработка комбинированного метода обнаружения и распознавания движущихся объектов// Приложение к журналу «Информационные технологии» №12/2013. Москва, 2013. С. 24-31.

НЕЧЕТКИЕ ЭЛЕМЕНТЫ В ЛОГИКО-МНОЖЕСТВЕННОЙ ГИСТОГРАММНОЙ МОДЕЛИ ПРЕДСТАВЛЕНИЯ ДАННЫХ

Папулин С.Ю.

Научный руководитель: к.т.н., доцент, Власов А.И.

МГТУ им. Н.Э.Баумана, Москва, Россия

FUZZY ELEMENTS IN LOGIC-SET MODEL OF DATA REPRESENTATION

Papulin S.Y.

Supervisor: Ph.D., Associate Professor, Vlasov A.I.

Bauman MSTU, Moscow, Russia

Аннотация

В статье рассматриваются вопросы использования нечетких элементов при анализе данных посредством логико-множественной гистограммой модели представления с применением элементного запроса. Обозначенные элементы способствуют разрешению проблемы использования неточных, неоднозначных определений элементов в модели представления, которые затем могут быть использованы в элементном запросе.

Abstract

This article considers matters of use of fuzzy elements for data analysis by means of logic-set histogram model of data representation using element queries. The elements resolve the problem associated with defining unclear, ambiguous model's elements that thereafter can be used in element queries.

Введение

Логико-множественная гистограммная модель представления – это средство анализа данных по запросу, позволяющее определить количественный показатель присутствия указанного в запросе элементного состава в данных за счет использования гистограммного представления данных [1]. Однако в стандартном варианте элементный состав E^G модели представления формируется за счет объединения базовых элементов (элементов универсального множества U) в различных сочетаниях. Другими словами, базовый элемент либо входит в состав элемента из множества E^G , либо нет без учёта степени их соответствия этому элементу. Данное обстоятельство зачастую затрудняет определение элементного состава E^G , особенно в тех случаях, когда нет однозначного представления о том, какие базовые единицы должны входить в состав элемента модели. Это, в конечном счете, приводит к тому, что при анализе, в том числе поиске, часть данных либо нецелесообразно учитывается системой, либо вовсе не принимается во внимание.

Для решения обозначенной проблемы предлагается воспользоваться нечеткими множествами, т.к. они применяются для описания различных видов неопределенностей, возникающих при обработке и анализе данных, а также позволяют принимать решения в условиях неполных или нечетких данных [2,3].

1 Нечеткое множество

Нечетким множеством A называют пару $(U, \mu_A(x))$, где U – универсальное множество, $\mu_A(x)$ – функция принадлежности нечеткого множества, определенная на множестве U и принимающая значения на отрезке $[0,1]$, $x \in U$ [2]. Оно имеет следующую форму записи:

$$A = \bigcup_{x \in U} \mu_A(x) / x, \quad (1)$$

где $\bigcup_{x \in U}$ – знак, обозначающий объединение элементов $\mu_A(x)$ в одно множество по всем $x \in U$; / – обозначение смыслового разделения функции принадлежности $\mu_A(x)$ и элемента универсального множества x ; $0 \leq \mu_A(x) \leq 1$.

Для упрощения дальнейшего изложения будем полагать, что U состоит из конечного числа элементов. Кроме того, введём взаимно-однозначную функцию $\text{indx}(x)$, которая каждому элементу U ставит в соответствие определенное натуральное число из множества $U' = \{i | i \in \mathbb{N}, 1 \leq i \leq |U|\}$.

С учётом вышеупомянутых замечаний, обозначим множество всех нечетких множеств в виде $E^{G,F}$. Тогда можно определить алгебраическую систему нечетких множеств следующим образом:

$$\mathcal{F} = \langle E^{G,F}, Op^F = \{\cup, \cap, \setminus, \overline{}\}; \mathbf{0}^F, \mathbf{1}^F \rangle, \quad (2)$$

$$\text{где } E^{G,F} = \{E | E = \bigcup_{i \in U''} \mu_{E^F}(i) / i, U'' \subseteq U', 0 \leq \mu_{E^F}(x) \leq 1, \sum_{x \in U''} \mu_{E^F}(x) \leq |U''|\} \quad (3)$$

или в виде упорядоченной последовательности

$$E^{G,F} = \left\{ \left(\mu_{E^F}(1), \dots, \mu_{E^F}(M) \right) \middle| 0 \leq \mu_{E^F}(i) \leq 1, \sum_{i=1}^M \mu_{E^F}(i) \leq |U'| = M \right\}; \quad (4)$$

Op^F – множество операций на множестве $E^{G,F}$; $\mathbf{0}^F$ и $\mathbf{1}^F$ – константы системы такие, что $\mathbf{0}^F = \bigcup_{i \in U'} 0 / i$, а $\mathbf{1}^F = \bigcup_{i \in U'} 1 / i$.

Операции «ОБЪЕДИНЕНИЕ», «ПЕРЕСЕЧЕНИЕ» и «ДОПОЛНЕНИЕ» имеют следующий вид, соответственно:

$$E_1^F \cup E_2^F = \bigcup_{i \in U'} \max(\mu_{E_1^F}(i), \mu_{E_2^F}(i)) / i, \quad (5)$$

$$E_1^F \cap E_2^F = \bigcup_{i \in U'} \min(\mu_{E_1^F}(i), \mu_{E_2^F}(i)) / i, \quad (6)$$

$$\overline{E^F} = \bigcup_{i \in U'} (1 - \mu_{E_1^F}(i)) / i, \quad (7)$$

где E_1^F , E_2^F и E^F – элементы множеству $E^{G,F}$.

Таким образом, нечёткие множества позволяют представить неоднозначные понятия в виде набора характеристических функций, значения которых определяют степень принадлежности элементов универсального множества понятию. При этом значение характеристической функции лежит на отрезке от 0 до 1.

2 Логико-множественное гистограммное представление данных

Для выполнения анализа данных посредством логико-множественной гистограммной модели представления описания данных должны иметь вид гистограммы, а критерии анализа необходимо задать в форме запроса. В общем виде гистограммное представление данных с учетом ранее введенных обозначений имеет следующий вид:

$$H^D = \left((H(1), \dots, H(M)) \middle| 0 \leq H(i) \leq \sum H^D, 1 \leq i \leq M, \sum_{i=1}^M H(i) = \sum H^D \right), \quad (8)$$

где M – количество элементов универсального множества U ; $\sum H^D = N^{DE}$, если гистограмма ненормализованная; $\sum H^D = 1$, если нормализованная гистограмма; N^{DE} – количество элементов в данных.

Для формирования элементного запроса используются элементный состав E^G модели представления и логические и/или множественные операции. Множество E^G в общем виде записывается следующим образом:

$$E^G = \{E | E \subseteq U\}. \quad (9)$$

Элемент E множества E^G можно представить в форме нечеткого множества следующим образом:

$$E = \bigcup_{i \in U'} \mu_E(i) / i \quad (10)$$

или в виде упорядоченной последовательности

$$E = (\mu_E(1), \dots, \mu_E(M)), \quad (11)$$

где $\mu_E(i) = 1$, если $\text{indx}^{-1}(i) \in E$; $\mu_E(i) = 0$ – иначе.

В результате элементный запрос имеет вид элементного высказывания:

$$S^E = \text{Высказывание}_E(E_1, \dots, E_L), \quad (12)$$

где $E_j \in E^G, 1 \leq j \leq L$; L – количество элементов высказывания.

Основной задачей при анализе данных с использованием логико-множественной гистограммной модели представления является определение количественного показателя присутствия элементного состава, обозначенного в элементном высказывании. Для этого элементное высказывание необходимо преобразовать в гистограммное S^H и вычислить его количественное значение ΣS^H .

Чтобы получить гистограммное высказывание, прежде всего, следует каждый элемент элементного высказывания перевести в гистограммный вид, а затем применить функцию преобразования. С учётом выражения (11) первое можно представить следующим образом:

$$H_j^E = \left((H(1), \dots, H(M)) \middle| H(i) = k \cdot E_j(i) \right), \quad (13)$$

где E_j – элемент S^E ; k – коэффициент равный 1 для нормализованной гистограммы и N^{DE} для ненормализованной.

Функция же преобразования выглядит для случая нормализованной гистограммы данных следующим образом:

$$SF^H(E_j) = \left((H(1), \dots, H(M)) \middle| H(i) = \min(H^D(i), H_j^E(i)) \right) = H_j, \quad (14)$$

где H_j – гистограммное представление E_j в контексте гистограммы данных H^D .

В результате в упрощенном виде гистограммное высказывание есть элементное высказывание, каждый элемент которого заменен на соответствующую ему гистограмму, полученную посредством выражения (14). Для вычисления значения гистограммного высказывания применяется алгебра гистограммного представления [1]:

$$\mathcal{H} = \langle H^G, Op^H = \{ \cup, \cap, \wedge, \vee, \dot{\vee}, \dot{\cap}, \overline{} \}; \mathbf{0}, \mathbf{1} \rangle, \quad (15)$$

где Op^H – операции на множестве H^G ; $\mathbf{0}$ и $\mathbf{1}$ – константы, где $\mathbf{0}$ – гистограммное представление, значения всех элементов которого равно нулю, а $\mathbf{1} = H^D$; H^G – множество гистограммных представлений элементов E^G в контексте гистограммы данных H^D , которое в общем виде записывается следующим образом:

$$H^G = \{ (H(1), \dots, H(M)) \mid 0 \leq H(i) \leq H^D(i); 1 \leq i \leq M, \sum_{i=1}^M H(i) \leq \Sigma H^D \}. \quad (16)$$

Количественный показатель присутствия есть количественное значение гистограммного высказывания, которое вычисляется как сумма его элементов [1].

Таким образом, анализ данных посредством логико-множественного гистограммного представления данных – это сложный процесс взаимодействия всех вышеприведенных компонентов. Поэтому для внедрения нечетких множеств в качестве элементного состава необходимо учитывать особенности компонентов рассматриваемой модели представления.

3 Нечеткие элементы в логико-множественном гистограммном представлении данных

Для реализации возможности использования нечетко определенных элементов в запросе пользователя при анализе данных посредством логико-множественной гистограммной модели данных предлагается заменить элементный состав модели E^G (9) на множество

нечетких элементов $E^{G,F}$ (3). В частном случае множество $E^{G,F}$ может состоять из того же количества тех же элементов, что и E^G , только определенных в виде нечетких множеств. В результате элемент E^F множества $E^{G,F}$ запишем в виде:

$$E^F = \bigcup_{i \in U'} \mu_{E^F}(i) / i \quad (17)$$

или в виде упорядоченной последовательности

$$E^F = (\mu_{E^F}(1), \dots, \mu_{E^F}(M)), \quad (18)$$

где $\mu_{E^F}(i)$ – характеристическая функция, определяющая степень принадлежности элемента $\text{indx}^{-1}(i)$ элементу E , $0 \leq \mu_{E^F}(i) \leq 1$.

В результате элементное высказывание S^E будет состоять из нечетких элементов множества $E^{G,F}$, а выражения (13) и (14) определим следующим образом, соответственно:

$$H_j^{E,F} = ((H(1), \dots, H(M)) | H(i) = E_j^F(i)), \quad (19)$$

$$SF^{H,F}(E_j^F) = ((H(1), \dots, H(M)) | H(i) = H^D(i) \cdot H^{E,F}(i)) = H_j^F \in H^G. \quad (20)$$

Дальнейшее вычисление количественного показателя присутствия будет аналогично рассмотренному в предыдущем разделе для элементного состава в виде обычных множеств.

В то же время операции «ОТРИЦАНИЕ» ($\bar{}$) и «КРОМЕ» (\setminus) в алгебре \mathcal{H} (15) желательно переопределить с учетом специфики нечетких множеств. Поэтому операцию «ОТРИЦАНИЕ» на множестве H^G по аналогии с выражением (7) запишем в следующем виде:

$$\overline{H^F} = ((H(1), \dots, H(M)) | H(i) = H^D(i) - H^F(i)). \quad (21)$$

Операцию «КРОМЕ» определим следующим образом:

$$H_1^F \setminus H_2^F = \left((H(1), \dots, H(M)) \middle| H(i) = \begin{cases} H_1^F(i) - H_2^F(i), & \text{если } H_1^F(i) \geq 0, H_2^F(i) \geq 0 \text{ и} \\ H_1^F(i) \geq H_2^F(i); \\ 0, & \text{иначе} \end{cases} \right). \quad (22)$$

Отдельно следует отметить, что одной из особенностей при использовании системы нечетких множеств \mathcal{F} (2) является несоблюдение закона исключения третьего и закона противоречия. В результате эта особенность в случае с нечетким элементным составом перетекает и в логико-множественное гистограммное представление.

Заключение

Таким образом, основным достоинством предлагаемого подхода является возможность использования нечетких элементов в составе логико-множественной гистограммной модели представления для анализа данных по элементному запросу. В то же время для этого были переопределены, помимо элементного состава, выражение для определения гистограммы элемента, функция преобразования гистограммы элемента в контекст гистограммы данных, а также отдельные операции алгебры гистограммного представления данных.

Литература

1. Папулин С.Ю. Свойства бинарных операций логико-гистограммного представления данных // 15-ая Молодежная международная научно-техническая конференция «Наукоемкие технологии и интеллектуальные системы 2013». Том 1. – М.: МГТУ им. Н.Э. Баумана, 2013. С.118-124. Режим доступа: http://www.iu4.ru/2013_ts/02_tom01.pdf (дата обращения: 15.12.2013).
2. Кобышева Л.К., Назаров Д.М. Основы теории нечетких множеств: Учебное пособие. СПб.: Питер, 2011. 192с.
3. Матвеев М.Г., Свиридов А.С., Алейникова Н.А. Модели и методы искусственного интеллекта. Применение в экономике: учеб. пособие. М.: Финансы и статистика; ИНФРА-М, 2008. 448с.

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ АЛГОРИТМА ЭВОЛЮЦИИ РАЗУМА

Сахаров М.К.

Научный руководитель: д.ф.-м.н., профессор, Карпенко А.П.

МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

INVESTIGATION OF THE EFFICIENCY OF MIND EVOLUTIONARY COMPUTATION

Sakharov M.K.

Supervisor: Dr., Prof., Karpenko A.P.

MSTU, Moscow, Russia

Аннотация

В статье представлено описание модифицированного алгоритма эволюции разума (*MEC*); предложена его интерактивная программная реализация в среде *Wolfram Mathematica*. Приведены результаты исследования чувствительности и эффективности алгоритма с использованием различных тестовых функций. На основании полученных результатов даны рекомендации по выбору значений свободных параметров алгоритма.

Abstract

This article gives a brief description of the modified mind evolutionary computation method (*MEC*); the algorithm was implemented in *Wolfram Mathematica*. Results of sensitivity and efficiency investigations with a use of several benchmark functions were also presented in this work. Recommendations on the selection of values of the *MEC*'s free parameters were drawn on the basis of obtained results.

Введение

Одной из основных особенностей популяционных методов глобальной оптимизации является наличие большого числа свободных параметров. От значений этих параметров существенно зависит эффективность методов, но несмотря на это, как правило, отсутствуют рекомендации по выбору значений этих параметров, которые позволили бы максимально эффективно решать различные классы задач оптимизации. Работа посвящена разработке таких рекомендаций для модифицированного варианта алгоритма эволюции разума (*MEC*), предложенного авторами в работе [1]. В этой работе модифицированный *MEC* использовался в качестве алгоритма верхнего уровня при разработке адаптивного мультимемеевого алгоритма поисковой оптимизации.

В рамках исследования была создана программная реализация модифицированного алгоритма эволюции разума, ориентированная на интерактивное изменение значений свободных параметров. Было проведено исследование чувствительности алгоритма к изменению значений его различных свободных параметров. По результатам исследования определены параметры, оказывающие наибольшее влияние на эффективность алгоритма, а также оптимальные значения остальных параметров. Полученные данные использованы для широкого исследования эффективности алгоритма с использованием многомерных многоэкстремальных и овражных тестовых функций.

1 Модифицированный MEC

Подробно постановка задачи и модифицированный алгоритм описаны в работе [1]. В данном разделе приведем лишь краткое описание основных шагов алгоритма и его свободных параметров.

Модифицированный *MEC* предназначен для решения задач безусловной глобальной оптимизации. Алгоритм описывается следующей последовательностью шагов.

1) Операция декомпозиции разбивает область поиска на равные подобласти с целью предотвращения преждевременной сходимости алгоритма.

- 2) Операция инициализации в каждой из указанных подобластей создает некоторое число групп индивидов заданного размера.
- 3) Операция локальных состязаний осуществляет локальный поиск лучшего индивида в каждой группе.
- 4) Операция диссимилиации управляет глобальным поиском, выбирая на каждой итерации лучшую группу по значению ее лучшего индивида. Некоторое число отстающих групп удаляется на каждой итерации, а вместо них генерируются новые группы.

В качестве критерия останова используется стагнация вычислительного процесса. Расчет прекращается если наилучшее найденное решение не изменяется в пределах заданной точности в течение числа итераций $\lambda_{stop} = 100$.

Рассматриваемый алгоритм имеет большое число свободных параметров. Прежде всего, это общее число групп L , которое определяется как произведение числа подобластей декомпозиции η на число групп в одной подобласти γ . Параметр η может принимать значения от 1 до 40, а параметр γ – от 1 до 7. Численность группы $|S_i|$ принята одинаковой и равной $|S|$. Величина $|S|$ варьируется от 5 до 80. Параметр $|S^b|$, определяющий число лидирующих групп, принимает значения от 1 до L . Два последних свободных параметра – среднеквадратичные отклонения σ_1 и σ_2 , где σ_1 – среднеквадратичное отклонение в нормальном распределении при генерации новых индивидов вокруг главного индивида каждой группы, σ_2 – аналогичная величина, используемая при инициализации новых групп вокруг наиболее успешной группы после удаления некоторого числа отстающих групп.

2 Программная реализация

Исследуемый алгоритм реализован в среде *Wolfram Mathematica*, которая позволяет легко разрабатывать различные интерактивные приложения с использованием процедурной или функциональной парадигм программирования. На рис. 1, 2 изображен интерфейс созданного приложения.

Интерфейс состоит из двух таблиц. В первой с помощью различных объектов управления задаются значения свободных параметров алгоритма. Во второй таблице отображаются результаты вычислений в интерактивном режиме. Отображение в таком виде числа итераций стагнации позволяет оценить оставшееся время расчета, а значений компонентов вектора варьируемых параметров X – понять в какой части области поиска в данный момент находится лучший индивид.

В программной реализации предусмотрен автоматический генератор классических тестовых функций размерностью от 2 до 64. Для исследования использовались тестовые функции Растригина, Гриванка, Розенброка и Захарова. Первые две функции относятся к классу многоэкстремальных, а две последние – к классу овражных функций [2].

3 Исследование чувствительности

Исследование чувствительности проводилось с использованием 16-мерной сферы. Результаты исследования представлены в таблице 1. В верхней части таблицы представлены значения параметров для каждого эксперимента, в нижней части в левом столбце – значение варьируемого параметра, в правом – полученные значения (\bar{f} – среднее значение целевой функции по результатам 100 запусков, σ – среднеквадратичное отклонение).

	Границы области поиска	[-10, 10]
	Число областей декомпозиции	10
	Число групп в области	4
	Размерность задачи	16
	Численность группы	40
	Число лидирующих групп	30
	СКО	0.05
	СКО-2	0.2
	Точность	5
	Целевая функция	Sphere
<input checked="" type="checkbox"/> Запуск		

Рисунок 1 – Таблица для настройки входных параметров алгоритма

Лучшее найденное решение	0.00573204
Вектор x	{0.0307206, 0.00276913, 0.00231285, -0.0129763, 0.0046211, 0.0285343, -0.00505846, 0.0250159, 0.00324788, -0.0434101, 0.013888, -0.00585129, 0.00805332, 0.0258284, 0.0153439, -0.00551765}
Общее число итераций	277
Число итераций стагнации	67
Общее число групп	40

Рисунок 2 – Таблица с результатами работы алгоритма, обновляющаяся в процессе вычислений

Из результатов исследования видно, что найденные значения целевых функций в большинстве случаев не являются минимумами. В экспериментах 1 - 3 были получены ожидаемые результаты. Увеличение каждого из указанных параметров приводило к небольшому улучшению общего результата. В экспериментах 4, 5 изменение значений σ_1 и σ_2 приводит к существенному улучшению результатов.

Таблица 1 – Результаты исследования чувствительности

Эксперимент 1		Эксперимент 2		Эксперимент 3		Эксперимент 4		Эксперимент 5	
$L = 40, \eta = 10, S^b = 20, \sigma_1 = 1, S = 40, \sigma_2 = 1;$		$L = 40, \gamma = 4, \eta = 10, S^b = 20, \sigma_1 = 1, \sigma_2 = 1;$		$L = 40, \gamma = 4, \eta = 10, S = 40, \sigma_1 = 1, \sigma_2 = 1;$		$L = 40, \gamma = 4, \eta = 10, S^b = 30, S = 40, \sigma_2 = 1;$		$L = 40, \gamma = 4, \eta = 10, S^b = 30, S = 40, \sigma_1 = 0,5;$	
$\gamma = 2$	$\bar{f} = 2,95$ $\sigma = 0,59$	$ S = 20$	$\bar{f} = 2,67$ $\sigma = 0,53$	$ S^b = 10$	$\bar{f} = 3,02$ $\sigma = 0,54$	$\sigma_1 = 1$	$\bar{f} = 2,49$ $\sigma = 0,45$	$\sigma_2 = 1$	$\bar{f} = 0,54$ $\sigma = 0,09$
$\gamma = 4$	$\bar{f} = 2,64$ $\sigma = 0,51$	$ S = 40$	$\bar{f} = 2,35$ $\sigma = 0,38$	$ S^b = 20$	$\bar{f} = 2,70$ $\sigma = 0,47$	$\sigma_1 = 0,5$	$\bar{f} = 0,67$ $\sigma = 0,14$	$\sigma_2 = 0,5$	$\bar{f} = 0,53$ $\sigma = 0,09$
$\gamma = 6$	$\bar{f} = 2,58$ $\sigma = 0,45$	$ S = 60$	$\bar{f} = 2,13$ $\sigma = 0,35$	$ S^b = 30$	$\bar{f} = 2,49$ $\sigma = 0,45$	$\sigma_1 = 0,05$	$\bar{f} = 0,007$ $\sigma = 0,001$	$\sigma_2 = 0,05$	$\bar{f} = 0,012$ $\sigma = 0,002$

В дальнейших экспериментах решено варьировать лишь среднеквадратичные отклонения, а остальные параметры принимать близкими к максимально допустимым в разработанной программе.

4 Исследование эффективности

Исследование эффективности проводилось на четырех тестовых функциях размерности $|X| = 16, 32, 64$. Из-за высоких размерностей функций мультистарт было решено ограничить $k = 20$ запусками. Все эксперименты проводились при следующих значениях свободных параметров: $\sigma_2 = 0,5$. Результаты представлены на рис. 3,4 в виде ящичковых диаграм, отображающих среднее полученное значение целевой функции и соответствующий интерквартильный размах.

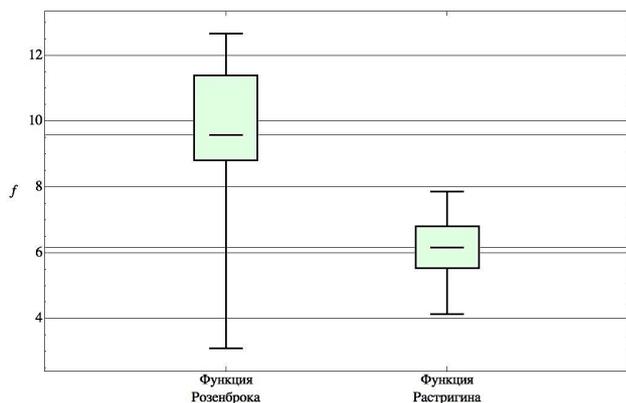


Рисунок 3 – Распределение значений целевой функции по результатам мултистарта: функции Розенброка и Растригина

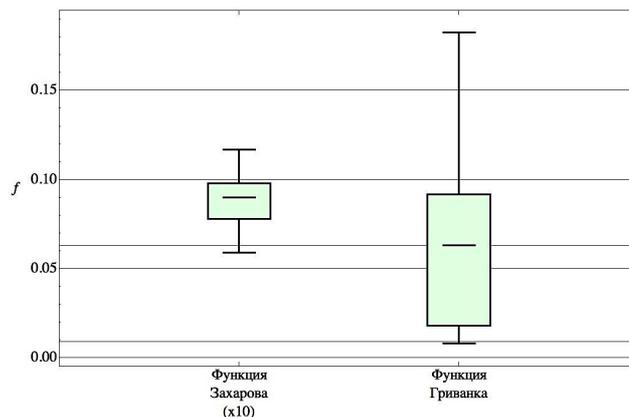


Рисунок 4 – Распределение значений целевой функции по результатам мултистарта: функции Гриванка и Захарова

Границы области поиска для всех функций кроме функции Растригина были приняты равными. В силу большого числа локальных оптимумов для функции Растригина область D была уменьшена до области

Заключение

Представленные результаты исследования эффективности алгоритма *MEC* показывают, что шестнадцатимерные функции Растригина и Розенброка оказались слишком сложными для данного алгоритма при относительно небольшом общем числе индивидов. Для функций Захарова и Гриванка глобальный минимум был локализован с невысокой точностью. Следует помнить, что при высокой размерности тестовых функций даже небольшие отличия значений элементов вектора X от известных оптимальных приводят к существенному изменению значений целевых функций.

Полученные результаты подтверждают необходимость предложенной авторами концепции меметического алгоритма эволюции разума [1] для повышения эффективности поисковой оптимизации. Высокие вычислительные затраты на оптимизацию многомерных тестовых функций указывают на необходимость реализации алгоритма на параллельных вычислительных системах [3].

Литература

1. Karpenko A., Posypkin M., Rubtsov A., Sakharov M. Multi-memetic Global Optimization based on the Mind Evolutionary Computation // Proceedings of the IV International Conference on Optimization Methods and Application "Optimization and applications" OPTIMA-2013. – M. 2013, pp. 83-84.
2. Сахаров М. К. Разработка и исследование гибридного мултимеметического алгоритма эволюции разума // Научно-технические технологии и интеллектуальные системы 2013. – М.: МГТУ им. Н.Э. Баумана, 2013. – С. 137-140.
3. Evtushenko Y., Posypkin M., Sigal I. A framework for parallel large-scale global optimization // Computer Science - Research and Development, 2009, pp. 211-215.

ТЕХНОЛОГИЯ ADO.NET

Шангитбаев Н.К.

Научный руководитель: к.т.н., доцент Курмангалиев Е.К.

Западно-Казахстанский инженерно-гуманитарный университет, Актобе, Казахстан

TECHNOLOGY ADO.NET

Shangitbayev N. K.

Supervisor:Ph.D., Ass.Prof. Kurmangaliyev E.K.

West Kazakhstan engineering humanities university, Aktobe, Kazakhstan

Аннотация

В статье дается обзор новой технологии ADO.NET на платформе .NET Framework.

Abstract

The article provides an overview of the new technology ADO.NET on the platform .NET Framework.

Введение

Сегодняшний век высоких технологий и научных достижений диктует высокий темп развития всех отраслей и наук, что привело к автоматизации большинства сфер жизни современного человека. Как у и любого процесса у автоматизации есть и положительные и отрицательные стороны. Одну из этих сторон, а именно упрощает ли автоматизация трудовую деятельность людей, рассмотрим в дальнейшем подробнее. С точки зрения любого руководителя автоматизация уменьшает издержки на производство и повышает эффективность работоспособности всей организации в целом. Но с точки зрения людей, которые разрабатывают системы автоматизации ситуация достаточно сложна, так как в основе любой информационной системы лежит база данных, а точнее СУБД. И выбор той или иной СУБД будет влиять на дальнейшие функциональные возможности информационной системы и то, как она будет спроектирована. А на сегодняшний день насчитывается около двух десятков различных СУБД. И перед разработчиком стает сложный выбор, какую СУБД использовать. А знать особенности проектирования при каждой СУБД непосильная задача даже для целого отдела разработчиков.

Ситуация еще усугубляется когда необходимо обеспечить поддержку различных источников данных. При чем каждый из таких источников данных может хранить и обрабатывать данные по-своему. Еще необходимо учитывать, что в различных языках программирования различна поддержка работы с той или иной СУБД.

То есть, еще возникает проблема не соответствия обработки информации большинством СУБД и способом обработки информации различными языками программирования.

Решение выдвинутых проблем найдено в новой технологии ADO.NET, разработанной компанией Microsoft, и включенной в их новую платформу .NET Framework [1].

Все проектировщики информационных систем подвержены одной большой проблеме: сложность выбора СУБД и дальнейшая реализация взаимодействия с ней. В связи с этим, целью данной работы является упрощение процесса проектирования ИС. Для реализации данной цели поставлена задача - разработать архитектуру, которая обладает возможностью масштабирования, адаптации к любому источнику данных. Архитектура должна быть проста в понимании разработчикам ИС, и обладать гибким механизмом использования ресурсов. Для реализации данной системы предлагается использовать технологию ADO.NET и платформы .NET.

1 Технология ADO.NET на платформе .NET Framework

ADO.NET - это часть Microsoft .NET Framework, т.е. набор средств и слоев, позволяющих приложению легко управлять и взаимодействовать со своим файловым или серверным хранилищем данных. ADO.NET - это модель доступа к данным в приложениях .NET. Ее можно использовать для доступа как к реляционным базам данных, таким как Microsoft SQL Server 2000, так и к другим источникам данных, для которых существуют OLE DB провайдеры. С одной стороны, ADO.NET представляет просто очередной этап развития технологии ADO, но, с другой стороны, ADO.NET предлагает фундаментальные нововведения, направленные на развитие в первую очередь Web-приложений, для которых характерно использование слабо связанных, фактически оторванных от источников наборов данных.

ADO — это набор библиотек, поставляемый с Microsoft .NET Framework и предназначенный для взаимодействия с различными хранилищами данных из .NET приложений. Библиотеки ADO.NET включают классы, которые служат для подсоединения к источнику данных, выполнения запросов и обработки их результатов. ADO.NET можно использовать и в качестве надежного иерархически организованного отсоединенного кэша данных для автономной работы с данными.

В NET Framework библиотеки ADO.NET находятся в пространстве имени System.Data. Эти библиотеки обеспечивают подключение к источникам данных, выполнении команд, а также хранилище, обработку и выборку данных (рис 1).

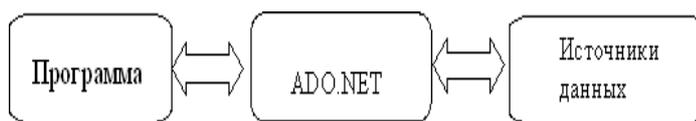


Рисунок 1 – Библиотека ADO.NET

ADO.NET отличается от предыдущих технологий доступа к данным тем, что она позволяет взаимодействовать с базой данных автономно, с помощью <отличенного> от базы кэша данных [2].

Автономный доступ к данным необходим, когда невозможно удерживать открытое физическое подключение к базе данных каждого отдельного пользователя или объекта.

Важным элементом автономного доступа к данным является контейнер для табличных данных, который не знает о СУБД. Такой незнающий о СУБД автономный контейнер для табличных данных представлен в библиотеках ADO.NET классом DataSet или DataTable.

Главный отсоединенный объект *DataSet* позволяет сортировать, искать, фильтровать, сохранять отложенные изменения и перемещаться по иерархичным данным. Кроме того, он включает ряд функций, сокращающих разрыв между традиционным доступом к данным и программированием с использованием XML. Теперь разработчики получили возможность работать с XML данными через обычные интерфейсы доступа к данным.

Технология ADO.NET призвана помогать разработке эффективных многоуровневых приложений для работы с БД в интрасетях и Интернете, для чего она и предоставляет все необходимые средства.

Объекты, составляющие отсоединенную часть модели ADO.NET, не взаимодействуют напрямую с подсоединенными объектами. В этом состоит их отличие от предыдущих объектных моделей доступа к данным Microsoft. В ADO объект *Recordset* хранит результаты запросов. Вы можете вызвать его метод *Open*, чтобы выбрать результаты запроса, а затем с помощью метода *Update* (или *UpdateBatch*) передать изменения из объекта *Recordset* в БД.

Объект *DataSet* ADO.NET (подробнее о нем рассказано далее) по функциональности сравним с объектом *Recordset* ADO. Тем не менее объект *DataSet* не взаимодействует с БД. Для выборки данных из БД и передачи их в объект *DataSet* последний передает методу *Fill* подсоединенного объекта ADO.NET — *DataAdapter*. Аналогично для передачи отложенных изменений из *DataSet* в БД объект *DataSet* нужно передать методу *DataAdapterUpdate* [3].

Заключение

Технология ADO.NET в полной мере способна предоставить механизм для доступа к любому источнику данных, тем самым, предоставляя разработчику мощный механизм взаимодействия с базами данных способный в полной мере реализовать все потребности, возникающие при проектировании ИС.

Литература

1. Троелсен Э. С# и платформа. NET. Библиотека программиста. – СПб.: Питер, 2004. – 86с.
2. Microsoft Corporation. Разработка Web-приложений на Microsoft Visual Basic. NET и Microsoft Visual С# .NET. Учебный курс MCAD/MCSD/Пер.с англ. – М.: Издательский дом «Русская Редакция», 2003. – 112с.
3. Нортроп Тони, Уилдермьюс Шон, Райан Билл. Основы разработки приложений на платформе Microsoft .NET Framework – М.:«Русская Редакция», 2007. – 92с.

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ ПРЕДПРИЯТИЕМ

Шангытбаева Г.А.

Научный руководитель: д.т.н., профессор, Ахметов Б.С.

Казахский национальный технический университет имени К.И.Сатпаева,
Алматы, Казахстан

AUTOMATED SYSTEMS CONTROL OF THE ENTERPRISE

Shangytbayeva G.A.

Supervisor: Dr., Prof., Akhmetov B.J.

Kazakh national technical university of a name of K.I.Satpayev
Almaty, Kazakhstan

Аннотация

В статье рассматриваются современные автоматизированные системы, ERP-системы и её особенности, программные системы, интегрируемые в рамках ERP-систем.

Abstract

The article considers enterprise resource planning and its features, the program systems which is integrated within ERP systems.

Введение

ERP-система (англ. Enterprise Resource Planning System — Система планирования ресурсов предприятия) — это интегрированная система на базе ИТ для управления внутренними и внешними ресурсами предприятия (значимые физические активы, финансовые, материально-технические и человеческие ресурсы). Цель системы — содействие потокам информации между всеми хозяйственными подразделениями (бизнес-функциями) внутри предприятия и информационная поддержка связей с другими предприятиями. Построенная, как правило, на централизованной базе данных, ERP-система формирует стандартизованное единое информационное пространство предприятия. ERP-системы – программное обеспечение, отвечающие за планирование, автоматизацию и мониторинг внутренних бизнес-процессов компании или отрасли [1].

1 Функции ERP-систем

В основе ERP-систем лежит принцип создания единого хранилища данных, содержащего всю корпоративную бизнес-информацию и обеспечивающего одновременный доступ к ней любого необходимого количества сотрудников предприятия, наделённых соответствующими полномочиями. Изменение данных производится через функции (функциональные возможности) системы.

ERP-система состоит из нескольких функциональных модулей, каждый из которых направлен на охват той или иной сферы деятельности предприятия (финансы, управление основными средствами, учет затрат и доходов предприятия, управление материальными запасами, используемыми в различных хозяйственных операциях, распределение, продажи, поставки, информация по реализации товаров, работ и услуг, людские ресурсы.)

В основе ERP-систем лежит принцип создания единого хранилища данных. Данное средство предназначено не только для сбора информации в одной точке из разрозненных источников. Специальная технология размещения данных позволяет также за считанные минуты обрабатывать информацию и, там, где для получения квартальных или годовых отчетов транзакционной ERP-системе потребуется несколько часов.

ERP помогает пользователям делать их работу более эффективно за счет устранения барьеров между бизнес-подразделениями. ERP-решение это:

- Предоставление полной картины в режиме реального времени, что помогает компаниям совершенствоваться за счет проактивного решения проблем;
- Улучшение финансового положения в соответствии с законодательными требованиями и сокращение рисков;
- Улучшение уровня обслуживания клиентов за счет предоставления одного источника для выставления счета [2].

2 Обзор облачных ERP-систем

В зависимости от размера компании и программного обеспечения система ERP может включать в себя модули для различных областей работы. Мировой рынок программного обеспечения класса ERP огромен и доминирующее положение на нем занимают такие гиганты как SAP, Microsoft, Oracle. Эти компании предлагают сложные комплексные решения с огромным количеством модулей и сложной программой внедрения и финансирования. В основном это программы устанавливаемые на большое количество рабочих станций и предъявляющие высокие требования как к оборудованию, так и к инфраструктуре.

Основные производители ERP-систем уверенно двигались в сторону облачных вычислений и в дальнейшем эта тенденция получит наибольшее развитие.

Многие предприятия различных отраслей промышленности все чаще используют «облачные» модели ИТ-решений. То есть, в последнее время компании всё чаще переходят на облачную модель использования решений, в том числе и ERP-систем. Облачная модель – это аренда ERP-системы через Интернет по безопасному удаленному каналу.

Облачные вычисления являются инновационной технологией, предоставляющей динамично масштабируемые вычислительные ресурсы и приложения посредством интернет-сервисов под управлением поставщика услуг с оплатой за реально полученные услуги или ресурсы. Облачные вычисления – относительно новый термин, на сегодняшний день уже прочно вошедший в общепринятую практику. Он используется для обозначения совокупности современных технологий, применяемых для распределенной обработки данных, при которой ресурсы вычислительных систем, программное обеспечение и информация предоставляются пользователю по запросу через сеть.

Облачные вычисления становятся все более популярными, особенно в последнее время, когда ограниченность финансовых ресурсов вынуждает компании оптимизировать затраты: не надо тратить огромные средства на создание собственных центров обработки данных, на оплату лицензионного программного обеспечения, на содержание квалифицированного персонала.

Применительно к ERP «облака» — это воплощение двух вещей. Первое — это внешний хостинг, что чаще всего диктует использование ERP через web-интерфейс. Второй не менее важный аспект — деньги. Покупая программный продукт, сразу инвестируется большая сумма. Пользуетесь SaaS-моделью, используется фактически лизинговая модель. В настоящий момент в России не существует полноценного рынка лизинга на лицензии ПО. При покупке необходимо сразу или в рассрочку (в зависимости от поставщика) приобрести лицензии. Кроме того, попутно приобретается все необходимое: сервер, платформенное ПО, в том числе операционную систему, СУБД и пр. В рамках облака предоставляется также и платформа, на которой работает ваше программное обеспечение. В этом случае необходимо платить только арендную плату [3].

Под облаком принято понимать единый с точки зрения клиента виртуальный сетевой узел, реализующий вычислительные службы (один или несколько серверов), который физически может представлять собой географически распределенную совокупность взаимосвязанных аппаратных узлов компьютерной сети. Облака — это ряд сервисов, которые выполняются и хранят данные у сервис-провайдеров. Таким образом, можно переместить свою локальную установку ERP-системы на сторону провайдера. К ней будет организован удаленный доступ от пользователей, которые находятся в вашем офисе. В

широком смысле «облако» означает именно набор сервисов — не отдельно взятая программа, а программа, которая интегрирована в другие сервисы.

Концепция «облачных» технологий медленно, но уверенно пробирается и в область решений для управления предприятием. Практически у любого серьезного поставщика, будь то SAP, Microsoft, Oracle, существует свои правила организации работы с системой посредством «облачных» технологий. Обладая экспертизой работы с решениями линейки Microsoft Dynamics, автор построит публикацию на основе принципов этих продуктов. Под поставщиком подразумевается организация, поставляющая услуги облачных вычислений, под потребителем – организация или физическое лицо, приобретающее такие услуги, а под пользователем – физическое лицо (сотрудник, партнер, гипотетический субъект – аппаратный или программный модуль), непосредственно использующее услуги, предоставленные поставщиком потребителю. При этом регулирование организации безопасности облачных вычислений и данных осуществляется на основании договора о предоставлении услуг (SLA – Service Line Agreement), заключаемого между поставщиком и потребителем. В зависимости от вида предоставляемых услуг варьируется и распределение ответственности поставщика и потребителя в вопросах обеспечения безопасности вычислений [4].

У поставщика Microsoft тема «облачных» вычислений последнее время занимает одно из ключевых мест в стратегии компании. Сегодня практически все основные продукты в том или ином виде можно использовать как в «классическом» виде, так и в «облачном». К последним относится также аренда программного обеспечения. Аренда по своей сути не совсем «облако», но обладает рядом присущих ему характеристик, например, удаленное размещение сервера, на котором инсталлирована система, а также возможность изменять количество лицензированных подключений в зависимости от потребностей компании. При этом подразумевается, что на удаленном сервере под правило аренды попадают также сопутствующие продукты, необходимые для работы ERP-системы: Microsoft Windows Server, Microsoft SQL Server, Microsoft Windows Remote Desktop, Microsoft Office. Последний служит для формирования отчетов и документов, и, кстати, пока используется классический вариант. А вот полностью «облачный» пакет Microsoft Office 365 планируется использовать тогда, когда появятся полностью разработанные в «облачных» технологиях ERP-системы Microsoft Dynamics [5].

При внедрении ERP системы на «облачном» сервере:

Достоинства:

1. И для фирмы внедренца, и для клиента нет необходимости в покупке дорогостоящего оборудования (серверы, шкафы, ИБП, кондиционеры, серверная комната и т.д.) - снижаются риски при старте новых проектов/направлений.

2. Можно очень гибко менять количество и мощность серверов, подстраивая под текущие нужды.

3. Если внедряемая ERP система использует веб интерфейс, проблемы подключения географически удаленных пользователей в простом варианте сводятся к подключению их к интернету. (То есть все пользователи могут работать с любого компьютера с доступом к сети, если иное не установлено политиками безопасности - например, доступ только с определенных IP адресов и т.п.)

4. Минимизируется риск «физического изъятия сервера». Этот плюс в основном имеет отношение только к России, Украине и подобным странам, но для многих предпринимателей из этих стран является очень важным. Как ни защищай сервер, устанавливая файрволы, ИБП и бронированные двери, всегда существует риск, что придут люди из «компетентных органов» и устроят «маски-шоу». В результате не только конфиденциальные данные получат более широкое распространение, чем хотелось, но и работа фирмы будет парализована на неопределенное время. И даже если в дальнейшем и удастся в суде доказать незаконность подобного мероприятия, ущерб уже будет нанесен.

Недостатки:

1. Требуется достаточно быстрый и стабильный доступ к Интернет для всех пользователей и внедренцев. (Сейчас в большинстве крупных городов доступен интернет по 3G и/или WiMAX - покупаешь модем, объединенный с WiFi - и проблема решена, срок установки и настройки - один день.)

2. Потенциально более высокая угроза хакерских атак.

3. Не все системы одинаково хорошо подходят для такой установки - желательно, чтобы ERP система имела веб интерфейс и система защиты ERP (лицензии) не мешала ставить на облачных серверах (Разумеется, можно использовать облачный сервер как обычный терминальный сервер, но ряд преимуществ будет потерян - при веб интерфейсе можно работать с любого устройства с доступом к интернет и браузером без дополнительной настройки. С системой защиты аналогично - например, большинство версий IC можно поставить или с большими проблемами, связывая облачный сервер и физический в локальную сеть, чтобы были видны ключи, или используя незаконные эмуляторы и/или патчи) [6].

Заключение

В общем объеме производимых вычислений неуклонно растет доля той совокупности современных технологий, которая получила броское наименование «облачные вычисления». Если раньше через Интернет были доступны, по большей части, лишь приложения, ответственные за обмен электронными сообщениями и публикацию web-страниц, то сегодня глобальная сеть все чаще используется для работы со многими другими программными приложениями и базами данных.

Для дальнейшего развития распределенных сетевых приложений и концентрации вычислительных ресурсов все более важной становится проблема обеспечения информационной безопасности. Использование облачных вычислений влечет за собой не только значительные экономические преимущества, такие как снижение затрат, оптимизация структуры инвестиций, повышение защищенности данных и перенос ответственности за обеспечение безопасности на поставщика услуг, но и значительные риски с точки зрения обеспечения информационной безопасности.

Рассмотренные виды услуг облачных вычислений и основных рисков, возникающих при их использовании, среди которых можно выделить организационные (такие как зависимость от поставщика услуг, невозможность соблюдения новых требований, ограничение контроля над используемыми службами) и технические (такие как нарушение изоляции данных, эксплуатация уязвимостей системы облачных вычислений, истощение ресурсов и отказ в обслуживании, несовместимость используемых разработок), лежат в основе рекомендаций для перехода на облачные технологии.

Литература

1. Сычев А.В. Теория и практика разработки современных клиентских веб-приложений. Интернет-Университет Информационных Технологий [Электронный ресурс]. Режим доступа: http://www.intuit.ru/department/internet/thpdevweba/24/thpdevweba_24.html, (дата обращения: 04.12.2013).
2. Молина Е., Хана С. Безопасность облачных вычислений: есть вопросы? [Электронный ресурс]. Режим доступа: <http://cloudzone.ru/articles/analytics/11.html> (дата обращения: 27.11.2013).
3. XaaS Check 2010 Status Quo und Trends im Cloud Computing. XaaS Check [Электронный ресурс]. Режим доступа: http://www.xaas-check.eu/download.php?cat=00_Willkommen&file=2010-XaaS-Check-Report.pdf, (дата обращения: 04.12.2013).
4. Волкова В.Н., Денисов А.А. Основы теории систем и системного анализа. – СПб-СПГТУ, 2001. – 42 с.
5. <http://zvukovaya-volna.ru> (дата обращения: 10.12.2013).
6. <http://www.it.ua/news.php> (дата обращения: 10.12.2013).

ШАБЛОНИЗАТОРЫ В WEB-ПРОГРАММИРОВАНИИ

Шевченко А.С.

Научный руководитель: к.т.н., доцент, Пивоварова Н.В.
МГТУ им. Н.Э.Баумана, кафедра РК6, Москва, Россия

TEMPLATE ENGINES IN WEB-PROGRAMMING

Shevchenko A.S.

Supervisor: Associate Professor, Ph.D., Pivovarova N.V.
BMSTU, CAD department, Moscow, Russia

Аннотация

В статье рассматриваются шаблонизаторы. Подробно исследована их классификация. Кратко освещены основные возможности. В заключении представлены рекомендации по выбору шаблонизатора для проекта.

Abstract

In the article Web template engines are viewed. Classification of them is investigated in detail. Their main possibilities are covered briefly. Recommendations on the choice of templates are represented in conclusion.

Введение

В настоящее время широко развито Web-программирование. В целях упрощения разработки сложных систем все большее применение находит схема MVC (Model-View-Controller) (рисунок 1). Данная схема имеет целью разделение модели данных, пользовательского интерфейса и взаимодействия приложения с пользователем на три отдельных компонента:

- Model – данные и правила для работы с ними;
- Controller – связь между пользователем и системой;
- View – отображение информации.

В результате модификация одного из компонентов оказывает минимальное воздействие на остальные.

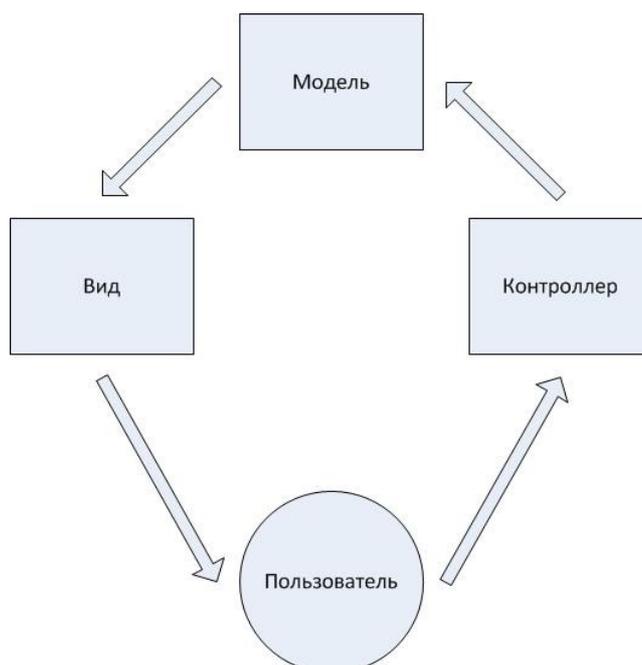


Рисунок 1 - Схема MVC (Model-View-Controller)

Для упрощения отделения пользовательского интерфейса (View) от остальных компонентов используются шаблонизаторы – программное обеспечение, позволяющее использовать Web-шаблоны для генерации конечных html-страниц.

Актуальность данного исследования состоит в том, что на текущий момент в источниках информации отсутствует подробная классификация и сравнение шаблонизаторов, и поиск таких сведений требует значительных усилий.

1 Классификация шаблонизаторов

Классификацию шаблонизаторов иллюстрирует рисунок 2.



Рисунок 2 - Классификация шаблонизаторов

Нативные шаблонизаторы. В качестве шаблонизатора в данном случае выступает язык, на котором написано само Web-приложение. С одной стороны, такие шаблонизаторы удобны для программиста, так как не требуют знания синтаксиса шаблонизатора. С другой стороны, при больших объемах кода они становятся сложными для восприятия, что затрудняет внесение изменений в проект. Такие шаблонизаторы используются в CMS, например, в Joomla. CMS представляет собой программное обеспечение для управления содержимым сайта и являются самыми быстрыми шаблонизаторами.

Интерпретируемые шаблонизаторы используют свой собственный синтаксис. Шаблоны, написанные с использованием такого подхода, требуют предварительной компиляции в нативные. По сравнению с нативными, код интерпретируемых шаблонизаторов менее сложен. В качестве примера таких шаблонизаторов можно привести Smarty и Twig. В публикациях их ещё называют «Smarty-подобные». Работают интерпретируемые шаблонизаторы немного медленнее нативных вследствие генерации менее компактного кода.

Блочные шаблонизаторы имеют отличный от нативных принцип работы. В них логика представления отделена от HTML-кода. Для шаблона используются два файла. В одном пишется HTML-код, в другом – логика представления, для написания которой применяется язык программирования, на котором написано Web-приложение. Логика представления решает такие задачи, как подключение других шаблонов, чередования и циклы при выводе и так далее. В качестве примера таких шаблонизаторов можно привести XTemplate. Блочные шаблонизаторы широкого распространения не получили в силу того, что необходимо создание дополнительного файла с логикой.

Шаблонизаторы XSLT (eXtensible Stylesheet Language Transformations) представляют собой язык преобразования XML-документов. В этих шаблонизаторах используется XLS-файл, задающий представление. Кроме того, используется XML-файл, задающий данные для вывода. При наложении указанных файлов происходит преобразование XML-файла в HTML-страницу. Данный шаблонизатор обеспечивает хорошую защищенность разработки, поскольку лицо, разрабатывающее интерфейс, не может внести изменений в алгоритм работы Web-приложения.

TAL (Template Attribute Language). В данном случае вставка команд и управляющих конструкций производится в атрибуты HTML-тегов. Подход TAL порождает самый короткий код по сравнению с другими шаблонизаторами. В силу специфичности подхода шаблонизаторы типа TAL сложны. Примером шаблонизатора данного типа является PHPTAL.

Помимо данной классификации, можно использовать и другие, например, по признаку подобия синтаксиса XML-коду. В таком случае TAL и XSLT можно отнести к XML-подобным шаблонизаторам.

2 Функциональность шаблонизаторов и примеры кода

Нативный шаблонизатор обладает всей функциональностью языка программирования, на котором написано Web-приложение (таблица).

Таблица 1 - Функциональность шаблонизаторов

	Переменные	Функции	Включения/ Условные включения	Циклы	Динамическое исполнение	Обработка ошибок	Наследование
Smarty	+	+	+/+	+	+	+	+
XTemplate	+(-)	+(-)	+/-	+(-)	+(-)	-	-
XSLT	+	+	+/-	+	-	-	+
PHPTAL	+	-	-/-	+	-	+	+

Под *включением* в таблице понимается включение содержимого одного шаблона в другой постоянно или по условию; динамическое исполнение означает выполнение кода, сохраненного в виде текстовых данных; наследование – подмена одного шаблона другим.

Пример кода Smarty:

```
<html>
<head>
  <title>{$title_text}</title>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
</head>
<body>
  <p>{$body_text}</p>
</body>
</html>
```

Пример кода XTemplate:

```
<!-- BEGIN:index -->
<!-- BEGIN:block -->
<p>{value}. Это содержимое блока </p>
<!-- END:block -->
<!-- END:index -->
```

В файле с логикой (в случае PHP): \$xtpl->assign('value', 2);

Пример кода XSLT:

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/TR/WD-xsl">
<xsl:template match="/">
<p><strong><xsl:value-of select="author"/></strong></p>
<p><xsl:value-of select="title"/></p>
</xsl:template>
</xsl:stylesheet>
```

Пример кода PHPTAL:

```
<div class="item" tal:repeat="value values">
  <div class="title">
    <span tal:condition="value/hasDate" tal:replace="value/getDate"/>
    <a tal:attributes="href value/getUrl" tal:content="value/getTitle"/>
  </div>
  <div id="content" tal:content="value/getContent"/>
</div>
```

Заключение

Выполнена классификация шаблонизаторов. Представлены примеры шаблонизаторов. Приведено краткое описание их функциональности, примеры кода.

На основании работы можно сделать вывод, что нет каких-либо универсальных рекомендаций по выбору шаблонизатора, при выборе нужно исходить из требований конкретного проекта, квалификации и состава коллектива разработчиков.

Литература

1. Хаген Г., Создание веб-сайтов с помощью Joomla! 1.5. — PacktPub, 2013. 296 с.
2. Мангано Сэл, XSLT. Сборник рецептов. — СПб.: БХВ-Петербург, 2008. 864 с.
3. Валиков А. Н., Технология XSLT. — СПб.: БХВ-Петербург, 2002. 544 с.

РАЗРАБОТКА ПРОГРАММНОЙ РЕАЛИЗАЦИИ КОМБИНИРОВАННОГО МЕТОДА ДЛЯ СОКРЫТИЯ ИНФОРМАЦИИ НА ОСНОВЕ ЦИФРОВОЙ ГОЛОГРАММЫ ФРЕНЕЛЯ И АЛГОРИТМА LSB

Спасёнов А.Ю.

Научный руководитель: к.т.н., доцент, Волосатова Т.М.
МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

DEVELOPMENT PROGRAMME IMPLEMENTATION COMBINED METHOD FOR EMBEDDING INFORMATION, USING THE DIGITAL FRESNEL HOLOGRAM AND LSB ALGORITHM

Spasenov A. Y.

Supervisor: Ph.D. assistant professor Volosatova T.M.
MSTU, Moscow, Russia

Аннотация

В статье рассматриваются алгоритмы стеганографии. Подробно исследован комбинированный метод сокрытия информации на основе цифровой голограммы Френеля и стеганографического метода LSB, представлена его программная реализация. В статье также приведены результаты работы созданного программного средства. На их основании проведено исследование возможности использования данного метода для защиты данных в САПР. В заключение представлены рекомендации по использованию данных алгоритмов.

Annotation

The algorithms of steganography are examined in the article. In it is in detail investigated the combined method of the concealment of information on the basis of the digital hologram of Fresnel and steganographic method LSB. Its program realization is represented. Examples of software's output were also included into the article. The possibility of applying these techniques to data protection in CAD systems was analyzed on the basis of these examples. Recommendations on the use of the specified algorithms were given in the conclusion.

Введение

Целью работы является разработка приложения для сокрытия информации на основе синтеза цифровой голограммы Френеля и метода LSB. В основе комбинированного метода заложено два этапа – получение цифрового голографического изображения защищаемых данных, а затем внедрение в контейнер методом LSB.

В рамках работы выполнены исследование методов стеганографии и их программная реализация. Разработано программное средство с графическим интерфейсом, позволяющее пользователю преобразовать изображение с помощью голографического метода, а также скрывать и извлекать информацию из изображений с использованием метода LSB. Результаты тестирования разработанного ПО, а также их анализ представлены в данной статье.

1 Постановка задачи

Стеганография – это расширяемая совокупность методов и средств передачи (хранения) скрытой информации, при которых скрытый канал организуется на базе и внутри открытого канала с использованием особенностей восприятия информации.

При проведении анализа принято считать, что основными стеганографическими понятиями являются сообщение и контейнер.

Контейнером будем называть несекретные данные, используемые для сокрытия сообщений. Сообщением будем называть секретные данные, наличие которых в контейнере необходимо скрыть. Ключом будем называть секретную информацию, известную только законному пользователю.

В данной работе сообщением будет являться модифицированное изображение, полученное на основе голограммы Френеля. Ключом будут являться параметры, с помощью которых будет восстанавливаться изображение. Контейнером будет являться изображение, в которое будет встроено сообщение, используя метод LSB.

2 Метод LSB

Метод LSB (Least Significant Bit, наименьший значащий бит) — суть этого метода заключается в замене последних значащих битов в контейнере (изображения, аудио или видеозаписи) на биты скрываемого сообщения. Разница между пустым и заполненным контейнерами должна быть не ощутима для органов восприятия человека.

Младший значащий бит изображения несет в себе меньше всего информации. Известно, что человек в большинстве случаев не способен заметить изменений в этом бите. Фактически, LSB – это шум, поэтому его можно использовать для встраивания информации путем замены менее значащих битов пикселей изображения битами секретного сообщения. При этом для изображения в градациях серого (каждый пиксель изображения кодируется одним байтом) объем встроенных данных может составлять 1/8 от общего объема контейнера.

Популярность данного метода обусловлена его простотой и тем, что он позволяет скрывать в относительно небольших файлах большие объемы информации (пропускная способность создаваемого скрытого канала связи составляет при этом от 12,5 до 30%). Метод, зачастую, работает с растровыми изображениями, представленными в формате без компрессии (например, BMP и GIF).

Метод LSB имеет низкую стеганографическую стойкость к атакам пассивного и активного нарушителей. Основной его недостаток – высокая чувствительность к малейшим искажениям контейнера.

3 Голограмма Френеля

3.1 Аналоговая голография Фурье и Френеля

Как известно, голография изначально представляла собой совокупность методов регистрации трехмерных изображений за счет кодирования координаты вдоль направления распространения электромагнитной волны в фазовой части.

Принципиальная схема голографической модели представлена на Рис. 1 и на Рис. 2.

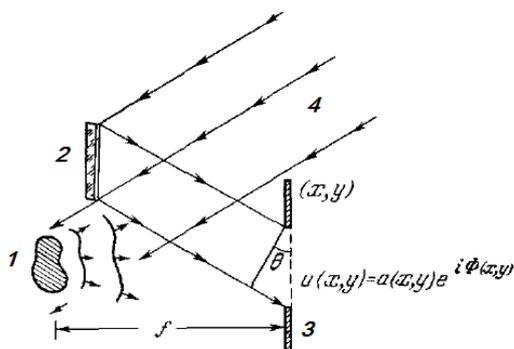


Рисунок 1 - Схема получения голограммы

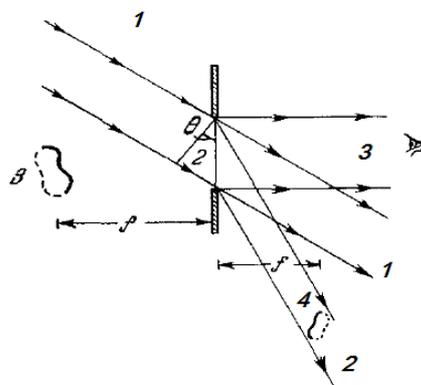


Рисунок 2 - Схема восстановления волны

- 1 – предмет.
- 2 - зеркало, создающее эталонный пучок.
- 3 - плоскость зрачка.
- 4 - когерентный источник света.

- 1 – эталонный пучок.
- 2 - голограмма.
- 3 – мнимое изображение предмета.
- 4 – действительное изображение предмета.

Голография, как правило, реализуется двухступенчатым процессом. На первом этапе фиксируется распределение интенсивности результата интерференции рассеянной волны от объекта и опорной волны, это и является голограммой. На втором этапе полученную голограмму используют как дифракционный транспарант. При этом, в зависимости от задачи, восстановленное поле строит изображение объекта или некий образ, характеризующий объект.

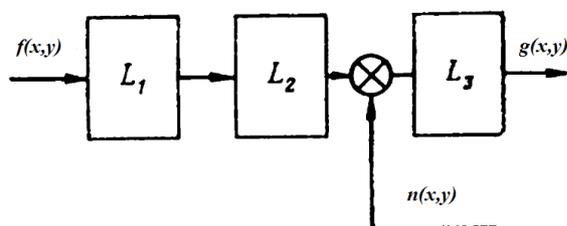


Рисунок 3 – Функциональная модель голографического процесса

Функциональная модель, приведенная на рисунке 3, описывает в общем виде задачу преобразования, записи голограммы и восстановления оптического изображения. Входной сигнал $f(x,y)$ распространяется от плоскости объекта до плоскости голограммы, чему соответствует система с оператором L_1 . При записи преобразованного сигнала $L_2[L_1[f(x,y)]]$ на голограмму накладываются шумы детектора $n(x,y)$. Для восстановления сигнала используют дополнительную систему L_3 .

3.2 Синтез цифровых голограмм Фурье и Френеля

Цифровые голограммы Фурье моделируют классические и представляют собой запись пространственного преобразования Фурье от рассеянного виртуальным предметом светового поля со сдвинутой виртуальной пространственной несущей. В общем случае алгоритм синтеза изображения ЦВЗ для получения голограммы $h(x,y)$ выглядит следующим образом:

- 1) В виртуальной предметной области (x,y) формируется изображение ЦВЗ $W(x,y)$;
- 2) ЦВЗ переносится в плоскость пространственных частот (u,v) со смещением относительно осей координат на значения M, N : $W(u-M, v-N)$;
- 3) Создается зеркальное изображение ЦВЗ и получается сумма двух изображений: $\mathfrak{Z}\{h(x,y)\} = \tilde{W}(u,v) = \gamma W(u-M, v-N) + \gamma W(u+M, v+N)$, где γ – коэффициент усиления;
- 4) Для получения исходного распределения $h(x,y)$ выполняется обратное преобразование Фурье: $h(x,y) = \mathfrak{Z}^{-1}\{\tilde{W}(u,v)\}$, где \mathfrak{Z}^{-1} – оператор обратного преобразования Фурье.

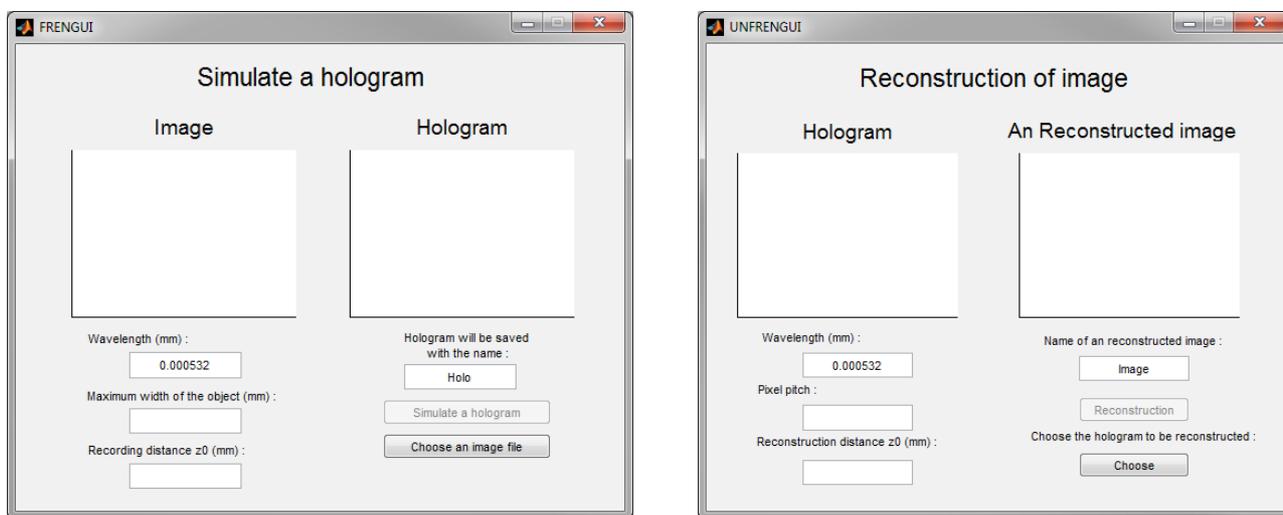
При моделировании голограммы Френеля процесс создания голограммы выглядит следующим образом:

- 1) В задаче синтеза голограммы естественно считать, что объект характеризуется достаточно гладкой функцией: $u(x,y)$
- 2) Дополнительный пучок, называемый опорным пучком, должен быть направлен под углом к плоскости пластинки: u_0
- 3) Задача синтеза голограммы Френеля сводится к расчёту матрицы $\{\Gamma(r,s)\}$
- 4) Исходная функция, характеризующая объект, может быть восстановлена по её

отсчётам путём интерполяции их некоторой функцией $\varphi(x,y)$

4 Концепция программной реализации

Программы реализованы в среде Matlab. На рис. 3 представлен графический интерфейс системы, позволяющей пользователю преобразовывать изображение, т.е. создать голограмму Френеля.

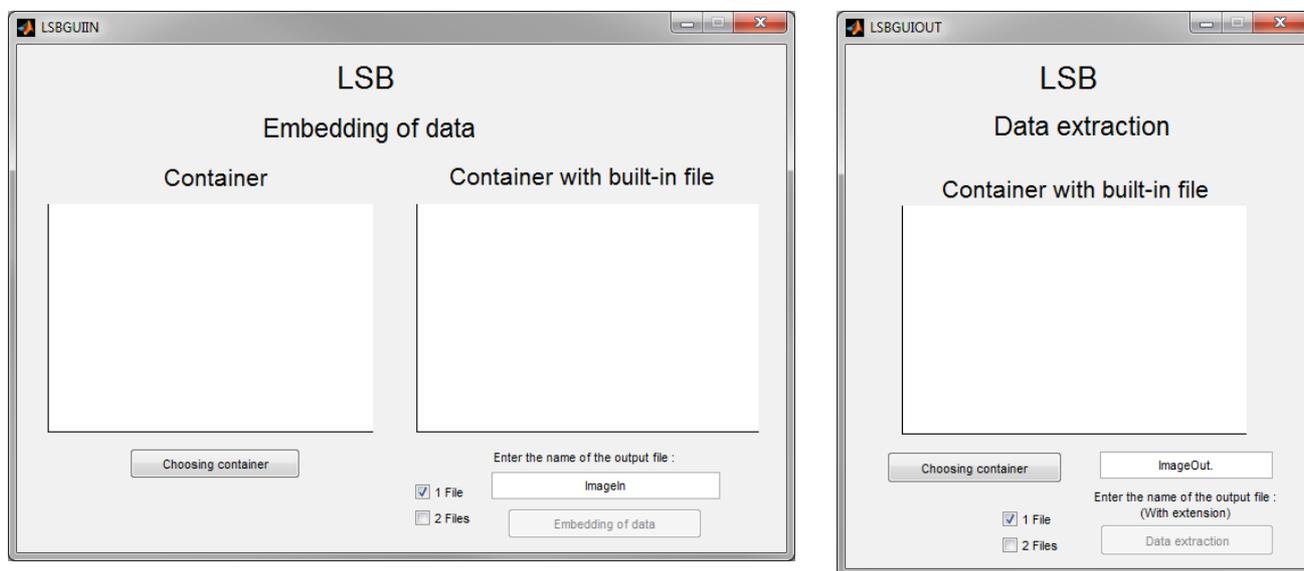


а) Интерфейс программы создания голограммы

б) Интерфейс программы восстановления изображения

Рисунок 4 – Графический интерфейс программы, реализующей голограмму Френеля

С помощью данных программы пользователь может модифицировать изображение, установив необходимые параметры (ключи) для моделирования голограммы Френеля.



а) Интерфейс программы встраивания данных

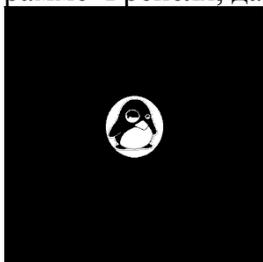
б) Интерфейс программы извлечения данных

Рисунок 5 – Графический интерфейс программы, реализующей метод LSB

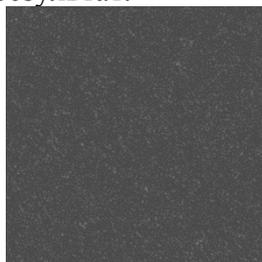
На рис. 5 представлен графический интерфейс программы, позволяющей пользователю скрыть данные в изображении. Для встраивания данных пользователь должен выбрать изображение, в которое они будут встраиваться.

5 Сравнение результатов

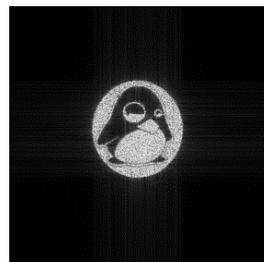
Из приведённых изображений видно, что алгоритм сокрытия данных, основанный на голограмме Френеля, даёт хороший результат.



а) Исходное изображение



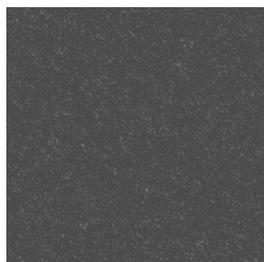
б) Модифицированное изображение
(Голограмма Френеля)



в) Восстановленное изображение

Рисунок 6 – Этапы создания голограммы и восстановления изображения на её основе.

Однако, непосредственное использование этого метода в проектной документации САПР приводит к порче файлов и потере данных (изображение может быть восстановлено, но некоторые фрагменты будут искажены). На рис. 7 представлены результаты работы программы встраивания данных в изображение на основе метода LSB.



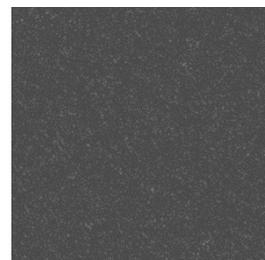
а) Встраиваемое изображение
(Сообщение)



б) Контейнер



в) Контейнер со встроенным сообщением



г) Извлеченное сообщение

Рисунок 7 – Этапы встраивания и извлечения данных, используя метод LSB

На основе результатов можно убедиться, что при замене наименее значимых битов очень трудно заметить наличие встроенного сообщения в изображение человеческим глазом.

Заключение

В результате проведения данной работы были разработаны программы для преобразования и внедрения информации на основе комбинированного метода стеганографии. Достоинством данной реализации является то, что даже если при передаче данных будет обнаружено наличие встроенного сообщения в изображение, понадобятся параметры, с помощью которых будет возможно восстановить изображение из голограммы. Это потребует дополнительных вычислительных и интеллектуальных ресурсов. К недостаткам можно отнести возможность удаления защищённой информации или её части из файла при передаче по сети интернет. В рамках данного проекта поставленные задачи были выполнены.

Литература

1. Т. М. Волосатова, А. В. Денисов, Н. В. Чичварин, Комбинированные методы защиты данных в САПР. - М.: Новые технологии, 2012. – 31с
2. Р. Гонсалес, Р. Вудс, С. Эддинс, Цифровая обработка изображений в среде MATLAB. - М.: Техносфера, 2006. – 616с.
3. И. Б. Бадриев, В. В. Бандеров, О. А. Задворнов, Разработка графического пользовательского интерфейса в среде MATLAB. – К., 2010. – 113с.
4. Poon T.-Ch., Optical scanning holography with MATLAB - Springer, 2007. – 157с
5. Р. Кольер, К. Беркхфрт, Л. Лин, Оптическая голография. – М.: Мир, 1973. – 698с.
6. Л. М. Сороко, Голография и интерференционная обработка информации. М., 1966. – 45с.
7. Н. К. Смоленцев, Создание Windows-приложений с использованием математических процедур MATLAB. - М.: ДКМ, - 2008. – 453с.
8. Ю. Кетков, А. Кетков, М. Шульц, MATLAB 7 программирование, численные методы. - С.-П.: БХВ-Петербург, 2005. – 753с.
9. Jun-chang Li, Pascal Picart, Digital Holography. - Wiley, 2012. – 358с

ЗАДАЧА КРЕДИТНОГО СКОРИНГА

Валиуллин А.М.

Научный руководитель: к.т.н., доцент, Волосатова Т.М.
МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

CREDIT SCORING PROBLEM

Valiullin A.M.

Supervisor: Ph.D., Associate Professor, Volosatova T.M.
BMSTU, Moscow, Russia

Аннотация

В статье обзорно рассматриваются различные задачи скоринга и методы их решения, основанные на численных статистических методах. Задача скоринга является одной из задач классификации машинного обучения, получившая своё развитие с появлением необходимости предсказания вероятности возврата кредита по некоторым данным заёмщика. Модель обучения построена по принципу «обучение с учителем», где данные для обучения берутся из анализа кредитной истории тысячи людей. Основные задачи скоринга заключаются в нахождении наиболее информативных признаков и построении наиболее приближенной модели.

Abstract

This paper introduces various scoring problems and methods of credit scoring that are based on numerical and statistical models. The scoring problem is also a classification problem of machine learning in essence. It's get developing with the emergence of the need to predict the default probabilities of credit takers. Learning model based on the principle of learning with a teacher. The data for training is taken from thousands of person's past credit history records. There are two general problems in scoring: finding the most informative features and constructing the most effective credit scoring model.

Введение

В странах с развитой системой финансовых услуг многие банки перед выдачей кредита подвергают заемщиков специальной процедуре, называемой скорингом. Задача скоринга является задачей классификации машинного обучения, где построение классификационной модели строится по принципу «обучение с учителем».

Идею разделения групп по вторичным признакам высказал в 1936 г. Английский статистик Рональд Фишер (R. A. Fisher) [1], предложив метод определения разновидностей ириса по результатам измерений частей растений. Несколькими годами позже в 1941 г. Давид Дюран (D. Durand) [2] предложил использование того же самого подхода для распознавания заемщиков с высокой и низкой вероятностью дефолта. Эта идея получила продолжение и в Сан-Франциско образовалась первая консалтинговая фирма Fair Issac Corporation, специализирующаяся на кредитном скоринге, которая и сейчас является одним из лидеров в разработке систем скоринга.

1 Виды скоринга

В современном мире во многом успешное функционирование различных кредитных организаций напрямую зависит от построения эффективных скоринговых систем. Под термином «скоринг» (от англ. «scoring» – подсчёт очков в игре) в широком смысле понимают различные методы получения вероятностной или количественной оценки заемщика. Скоринг базируется на различных финансовых, социально-демографических характеристиках клиентов. Сейчас различают несколько видов скоринга: кредитный скоринг (application scoring), где оценка вероятности возврата кредита происходит на основе некоторых его характеристик, в основном, содержащиеся в его заявке на получение кредитных средств; коллекторский скоринг (collection scoring), применяемый для работы с заемщиками, отказывающимися от возврата задолженности. Направлен для разработки

действий сотрудником кредитной организации, которые позволили бы вернуть «проблемный» заем. Поведенческий скоринг (behavioural scoring) позволяет оценить предполагаемые финансовые действия клиента по погашению кредита. Даёт возможность спрогнозировать изменения платежеспособности клиента для предупреждения возникновения задолженности. Скоринг мошенничества (fraud scoring) служит для оценки вероятности совершения мошеннических действий и выявления возможных мошенников среди лиц, претендующих на получение кредита. Это так называемый скоринг проверки благонадежности потенциального заемщика. Преимущества использования скоринга заключаются в сокращении времени и издержек, более тщательном анализе клиента и уменьшении возможного риска невозврата.

2 Задача классификации

Имеются исходные данные – выборка $D = \{(x_1, y_1), \dots, (x_i, y_i), \dots, (x_m, y_m)\}$, $x_i \in \mathbb{R}^n$, $y_i \in \{1, 0\}$ (возврат - невозврат кредита). Далее для удобства введем множество индексов объектов: $i \in I = \{1, \dots, m\}$. Векторы признаков $X \in \mathbb{R}^{m \times n}$, $X = (x_1, \dots, x_i, \dots, x_m)$. Для признаков также удобно ввести множество индексов: $J = \{1, \dots, n\}$. Аналогично матрице признаков введем вектор ответов $y = (y_1, \dots, y_i, \dots, y_m)^T$. Задана функция потерь $L(\alpha(x), y)$ в виде матрицы штрафов

$$L(\alpha(x_i), y_i) = \begin{cases} 0, & \text{если } \alpha(x_i) = y_i \\ 1, & \text{если } \alpha(x_i) \neq y_i \end{cases} \quad (1)$$

Требуется построить алгоритм классификации $\alpha : X \rightarrow \{0, 1\}$, который доставляет минимум функционалу качества

$$Q(\alpha, X) = \frac{1}{N} \sum_{i=1}^N L(\alpha(x_i), y_i) \quad (2)$$

3 Обучающая выборка и её размер

Для получения классификационной модели выборка исходных данных D разбивается на две непересекающиеся выборки: обучающую L и контрольную T . На обучающей выборке происходит построение модели классификации. Это множество включает в себя входные x_i и выходные значения y_i данных. Выходные значения y_i используются для обучения модели. Контрольная выборка также содержит входные x_i и выходные значения y_i . Выходные значения применяются для оценки точности классификации полученной модели. Такой принцип обучения называется «обучение с учителем».



Рисунок 1 – Процесс построения скоринговой модели и ее применение

При разделении заданной выборки на обучающую и контрольную возникает проблема определения их размера. С одной стороны, чем больше обучающая выборка, тем лучше будет построен алгоритм классификации нашей модели, хотя при превышении определенного объема данных может возникнуть эффект переобучения. С другой стороны, чем больше контрольная выборка, тем более точным будет определение частоты ошибок нашей модели. Возникает проблема выбора соотношения обучающего и контрольного набора данных. В работе [3] произведен сравнительный анализ различных соотношений двух выборок и дано рекомендованное соотношение обучающей и контрольной выборки при достаточном количестве записей в таблицах – 2:1.

4 Методы классификации

В основе скоринговой модели, которая решает задачу классификации, могут лежать различные алгоритмы классификации, такие как:

- метод ближайшего соседа (kNN, k-nearest neighbour algorithm)
- байесовский подход (Bayesian methods)
- метод опорных векторов (SVM, support vector machine)
- линейная регрессия (LR, linear regression)
- логистическая регрессия (LM, logit model)
- генетические алгоритмы (GA, genetic algorithms)
- нейронные сети (NN, neural networks)
- классификация на основе дерева решений (CART, classification and regression tree)
- и другие методы.

Указанные методы могут применяться как по отдельности, так и в различных комбинациях. Основными критериями выбора скоринговой модели являются точность и интерпретируемость. Легко догадаться, что даже небольшое повышение точности классификации дает значительные приросты прибыли кредиторов, и это стимулирует продолжение работ по созданию более точных алгоритмов. Также на практике при внедрении скоринговой модели важной становится задача интерпретации полученных результатов. Существенное ограничение состоит в том, что полученный алгоритм классификации должен быть как можно более простым, понятным и должен допускать интерпретацию на естественный язык в терминах кредитного риска. Некоторые мощные алгоритмы (например, нейронные сети и генетические алгоритмы) могут находить более сложные нелинейные зависимости, но их интерпретация является не такой простой задачей. Таким образом, в задаче скоринга ищется компромисс между простотой алгоритма и его точностью. Например, нейронные сети показывают довольно неплохие результаты, но при интерпретации и попытках объяснить значения весовых коэффициентов возникают трудности. Поэтому эта модель не нашла широкого распространения в задачах потребительского кредитования. Но способность этой модели находить сложные зависимости способствует выявлению сложных закономерностей, что дает применение этой модели при определении мошенничества с кредитными карточками [4].

5 Оценка модели

Для оценки точности модели классификации используется кривая ошибок или ROC-кривая (ROC – Receiver operating characteristic) – графическая характеристика качества бинарного классификатора, зависимость доли верных положительных классификаций от доли ложных положительных классификаций при варьировании порога решающего правила. В таблице 1 показана матрица ошибок для обучающей и контрольной выборки.

Таблица 1 - Матрица классификации

Актуальные данные	Предсказанные данные	
	«плохие»	«хорошие»
«плохие»	<i>TN</i>	<i>FN</i>
«хорошие»	<i>FP</i>	<i>TP</i>

TP, *TN* – правильно классифицированные клиенты из классов «хорошие» и «плохие»,
FP – ошибочно классифицированные клиенты, которые принадлежат классу «хорошие», а модель классифицирует их как класс «плохие» (ошибка II рода),
FN – ошибочно классифицированные клиенты, которые принадлежат классу «плохие», а модель классифицирует их как класс «хорошие» (ошибка I рода).

Для классификатора можно определить характеристики качества:

- 1) Доля верных положительных классификаций (True Positive Rate, TPR) или чувствительность модели:

$$TPR = \frac{TP}{TP + FN}$$

- 2) Доля ложных положительных классификаций (False Positive Rate, FPR):

$$FPR = \frac{FP}{TN + FP}$$

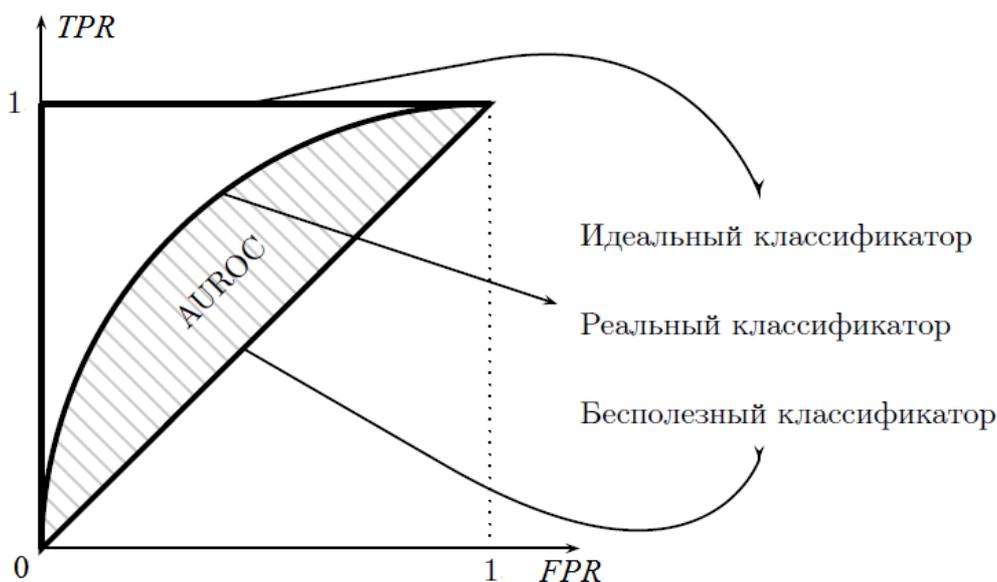


Рисунок 2 - ROC-кривая

ROC-кривая (рис. 2) показывает, что происходит с количеством ошибок I и II рода. ROC-кривая проходит из начала координат в точку (1,1). Чем выше лежит полученная ROC-кривая, тем лучше качество классификации. Идеальная ROC-кривая представляет собой «угол», проходящий через точки (0,0), (0,1) и (1,1). Наихудший алгоритм (случайный выбор) соответствует диагональной прямой $TPR = FPR$, соединяющей начало координат и точку (1,1), её также изображают на графике как ориентир [5]. Кроме графической характеристики, есть численная характеристика качества классификации AUC (Area Under Curve) – площадь под ROC-кривой.

Заключение

В работе подробно описана одна из задач классификации машинного обучения – задача скоринга, описаны различные скоринговые задачи, приведена постановка задачи кредитного скоринга, перечислены методы классификации, используемые для решения этих задач, описана кривая ошибок для оценки точности классификации. Скоринговые системы классификации заемщиков позволяют оперативно принимать решения о кредитовании, определять соотношение между прибыльностью организации и уровнем риска. В задаче кредитного скоринга основными критериями выбора модели являются точность полученной модели классификации и её интерпретация в терминах кредитного риска.

Литература

1. Fisher R. A. The use of multiple measurements in taxonomic problems // Annals of Eugenics, 1936, N. 7, 179-188 pp.
2. Durand D. Risk elements in consumer installment financing // NY: National Bureau of Economic Research, 1941.
3. Weiss S. M., Kulikowski C. Computer systems that learn: Classification and prediction methods from statistics, neural nets, machine learning, and expert Systems, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1991.
4. Syeda M., Zhang Y., Pan Y. Parallel granular neural networks for fast credit card fraud detection // Proceedings of the 2002 IEEE International Conference on Fuzzy Systems, 2002, 572 - 577 pp.
5. Воронцов К. В. Лекции по алгоритмам восстановления регрессии, 2007.
6. <http://www.ccas.ru/voron/download/Regression.pdf>

МЕХАНИЗМ МЕТАНИЯ МЯЧЕЙ С СИСТЕМОЙ АВТОМАТИЧЕСКОГО НАВЕДЕНИЯ НА ЦЕЛЬ

Вилкова Е.А.

Научный руководитель: Юдин А.В.

МГТУ им. Н.Э.Баумана, кафедра ИУ-3, Москва, Россия

BALL THROWING MECHANISM WITH AUTOMATIC TARGETTING SYSTEM

Vilkova E.A.

Supervisor: Yudin A.V.

Bauman Moscow State Technical University, Moscow, Russia

Аннотация

В статье рассматривается задача по разработке универсального пускового механизма для мобильного робота. В рамках данной работы проведено исследование правил соревнования Eurobot 2014 и подробно рассмотрена общая постановка задачи. Сделана модель пускового механизма, управление которым осуществляется системой автоматического наведения на цель. В заключение проанализирована эффективность предложенного решения в области механики и электроники.

Abstract

The article considers a problem of engineering a universal releaser for a mobile robot. In this work we study rules of the competition Eurobot 2014 and discuss in detail a general problem definition. Model of the releaser, controlled by the system of automatic guidance on target, is made. In conclusion, an efficiency of the proposed solution in such fields as mechanics and electronics is analyzed.

Введение

На сегодняшний день важным развивающимся направлением в области роботехники является, так называемое, «интеллектуальное» управление мехатронными модулями. Суть происходящего заключается в том, что накопленный потенциал в сфере электроники находит применение во все большем количестве мехатронных устройств. Микроминиатюризация всех компонентов подобной системы позволяет реализовывать практически любую задумку разработчика. Недавним примером в сфере уменьшения размеров электронных систем может служить анонсированный компанией Intel компьютер Edison, созданный по технологии Quark. Имея размеры SD-карты это полноценный компьютер с возможностью беспроводной связи, позволяющий расширить направление носимой электроники и дающий толчок к появлению новых и интересных решений [1].

В связи с возросшими техническими возможностями растет и возможность наделить устройства более «разумным» поведением – более развитыми и сложными алгоритмами, связывающими все возрастающее количество датчиков с не менее сложными в управлении исполнительными механизмами.

Таким образом, мехатронные технологии помогают отвечать таким современным требованиям к устройствам, как максимальная компактность конструкции, высокие скорости движения рабочих органов, «интеллектуальное» поведение системы, функционирующей в различных изменяющихся средах, способность системы к реконфигурации в зависимости от требуемых операций, надежность и безопасность функционирования.

Наиболее интересным для автора видится направление объединения мехатронных модулей в сложные системы и комплексы, допускающие быструю реконфигурацию, невысокую стоимость установки, настройки и обслуживания. Назначением мехатронных модулей движения является возможность управления движением (зачастую по одной координате). А совокупность таких узлов, скомбинированных в мехатронную систему, позволяют производить сложные многокоординатные управляемые перемещения.

Современные инженеры, конструируя устройства или сложные интегрированные системы, разбивают задачи на разные уровни. Это позволяет создавать различные элементы одного устройства параллельно, независимо друг от друга. Большинство таких элементов

являются базовыми и решают стандартные задачи в различного рода проектах. Создание таких элементов отнимают у разработчиков много времени, которое можно было эффективно использовать для оптимизации механизмов. Разработчик, зная модуль с описанными правилами работы, достоинствами и недостатками, может интегрировать этот механизм в создаваемое устройство в первоначальном виде или с некоторыми изменениями, в зависимости от задач проекта.

Международные соревнования мобильных роботов Eurobot проводятся ежегодно с целью вовлечения юных инженеров в робототехнику. В рамках этого конкурса требуется создание робота, выполняющего определенные задачи. Исправный робот представляет собой интегрированную систему модулей, каждый из которых отвечает за свою функцию. Отдельные механизмы, возможно, использовать в последующие годы в рамках этого и других соревнований. В данной статье описан один из таких «интеллектуальных» мехатронных модулей.

1 Постановка задачи

Правила соревнований Eurobot [2] каждый год меняются, и появляются новые задачи, которые команды должны воплотить в своих роботах. В рамках сезона 2014 года перед участниками поставили 5 различных задач.

Одной из таких задач является стрельба снарядами с робота в мишень «мамонта». Механизм, установленный на роботе, должен быть направлен на цель и снаряды брошены точно в небольшую область мишени на вертикально расположенном щите. Всего на игровом поле в разных местах расположены две мишени, покрытые липучкой Velcro.

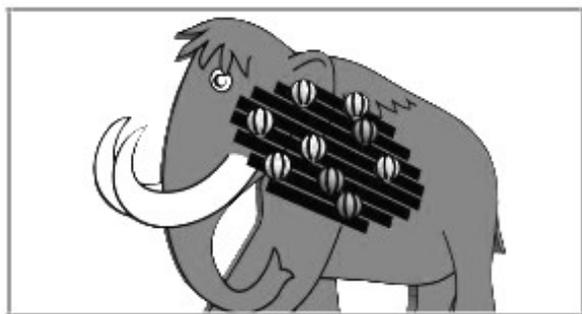


Рисунок 1 – Внешний вид мишени – «мамонта»

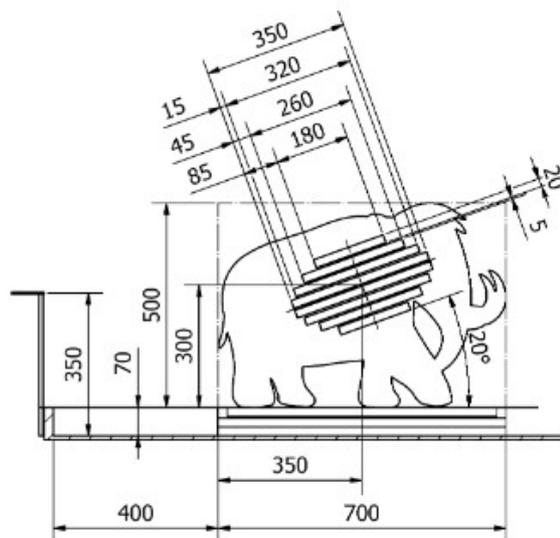


Рисунок 2 – Геометрические характеристики

По правилам соревнований снарядами служат 6 мячиков для настольного тенниса, покрытых липучкой Velcro. До начала матча все снаряды команды должны находиться на роботе для дальнейшего осуществления баллистического выстрела.

Соревнования Eurobot 2014 строго ограничивают участников в размерах робота в рабочем состоянии: периметр не более 1200 мм, а высота не более 350 мм. К тому же, система не должна повреждать робота соперников, мишени, игровую площадку, борт и другие элементы, находящиеся на поле. По правилам соревнований стоит учитывать ограничения на мощность систем, время работы батареи без подзарядки, длину используемых проводов и использование некоторых видов энергии.

2 Решение задачи в общем виде

В реализации мехатронного модуля первую и неотъемлемую часть занимает проработка механики. Проблема выбора пускового механизма для метания мячей была разрешена в пользу выстрела снаряда через дуло «пушки» (по аналогии с пушкой стреляющей ядрами или танковой пушки). Всем известно, что пушка является жесткой направляющей, позволяющей точно и просто управлять траекторией полета снаряда с помощью ее поворота.

Ниже на Рис. 3 представлен последний прототип созданного пускового механизма. Ранее были созданы 2 модели пушек упрощенного типа, необходимые для проведения экспериментов и доводке отдельных элементов системы.

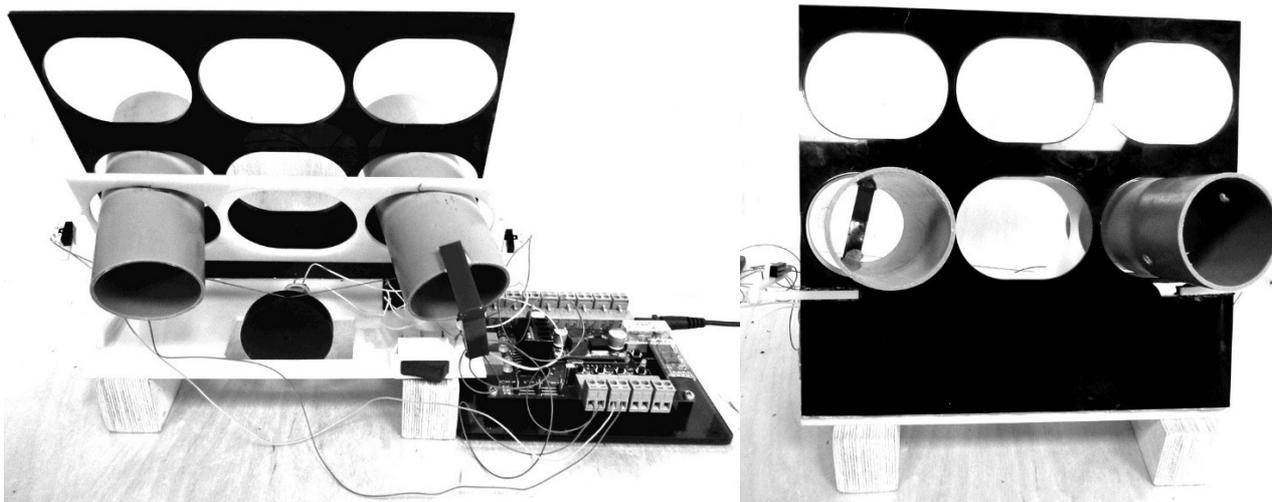


Рисунок 3 – прототип пускового механизма

Стоит отметить, что данная статья посвящена созданию мехатронного модуля, являющегося одним из элементов сложной системы. Именно поэтому мы рассматриваем наш механизм относительно существующего робота с другими элементами, системой движения и управления.

Следующим этапом реализации модуля является проработка электроники и системы автономного управления. Для этого необходимо выбрать архитектуру системы, состоящую из отдельных узлов, каждый из которых выполняет строго определенную функцию. Успех создания механизма видится в уменьшении количества узлов в системе без потери функциональности. Тогда созданный модуль будет наиболее простым, надежным, эффективным и универсальным. Структурная схема модуля представлена на Рис. 4.



Рисунок 4 – структурная схема модуля

Разработанный механизм можно разделить на 2 части или «этажа», каждый из которых содержит: систему из 3 труб, управляемую двигателем, 2 датчика-концевика и систему пуска мячиков, управляемую двигателем. Обе части конструктивно идентичны.

МИШЕНЬ 1, 2 – представляют собой элементы внешней среды и являются целью, которую должны достичь снаряды.

КАМЕРА – установлена на передней части робота для нахождения МИШЕНЕЙ. Она передает кадры на микроконтроллер МК.

МК – микроконтроллер, управляет всем механизмом.

ДАТЧИК 1, 2 – парные датчики-концевики для 1 и 2 этажа механизма, которые подают сигнал на МК в случае максимального отклонения труб от начального положения. Функция – повысить экономию энергии и безопасность системы за счет отключения двигателей, когда их работа не требуется, и вычислить время движения до амплитудных значений.

ДВИГАТЕЛЬ 1, 2 – двигатели 1 и 2 этажа мехатронного модуля, управляемые МК, которые позволяют направить трубы в нужном направлении.

СПУСК 1, 2 – пусковые механизмы для 1 и 2 этажа механизма, управляемые МК. Они состоят в том числе и из двигателей, которые приводят снаряды в движение.

3 Математические расчеты по применимости

В результате разработки модели пускового механизма стало необходимым проверить возможность применимости данного модуля в рамках соревнований Eurobot 2014. В данной статье описывается только один элемент, размеры которого 210 мм x 110 мм x 200 мм. По условию соревнований есть строгие ограничения на размеры всей системы. В исследовательских целях учитывается оптимальный по мнению автора размер робота (точная конфигурация робота зависит от многих факторов и может меняться до фиксации окончательной компоновки всех механизмов на его борту). Максимальная высота прописана в правилах (350 мм), а, зная предельный периметр поверхности, предположим, что в основании робота лежит квадрат со стороной 300 мм.

В пусковом механизме был на практике определен максимальный поворот направляющих труб, угол определяет возможность поражения мишеней механизмом. Наибольшее возможное отклонение от первоначального положения в каждую сторону составляет приблизительно 30 град. Помимо направления большую роль играет дальность полета снаряда. Эксперименты по метанию теннисного мячика из труб дали среднее значение полета, заключенное в размерах от 1000 мм до 1500 мм. На основании идеальных предельных значений и принятых габаритах робота проанализирована применимость данного модуля в системе в рамках соревнования Eurobot 2014. На Рис. 5 изображено игровое поле, на котором идет матч соревнований (вид сверху).

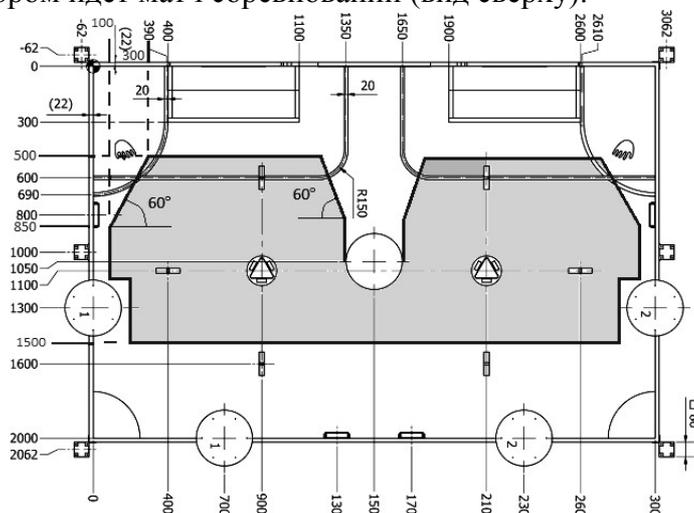


Рисунок 5 – игровое поле матча (вид сверху)

Выделенная темно-серая область показывает наиболее выгодное положение робота для выстрела в мишени.

4 Реверсивное движение

Управление выстрелом осуществляется механизмом поворота труб на необходимый угол. Поворот должен осуществляться в две стороны. Используется двигатель постоянного тока. Чтобы осуществить вращение вала двигателя, как в прямом, так и в обратном направлении, необходимо поменять полярность питания двигателя. Было решено создать схему, обеспечивающую реверсивное движение механизма наведения пушки. Эту задачу получилось осуществить с помощью 2 реле Song Chuan 793-P-1C, схема которых представлена на Рис. 6.

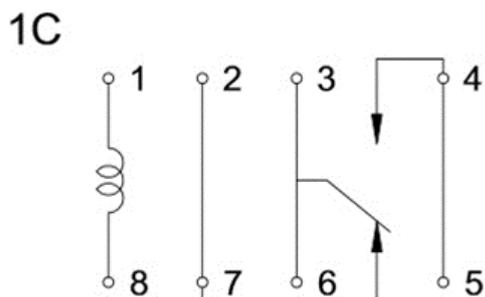


Рисунок 6 – схема реле

В начальном положении схема работает в последовательности 2-7-3-6. При подаче напряжения через катушку 1-8 ток идет в последовательности 3-6-4-5. Используя это свойство, была спаяна электронная схема, позволяющая осуществить реверсивное движение. Схема решения данной задачи представлена на рисунке 7.

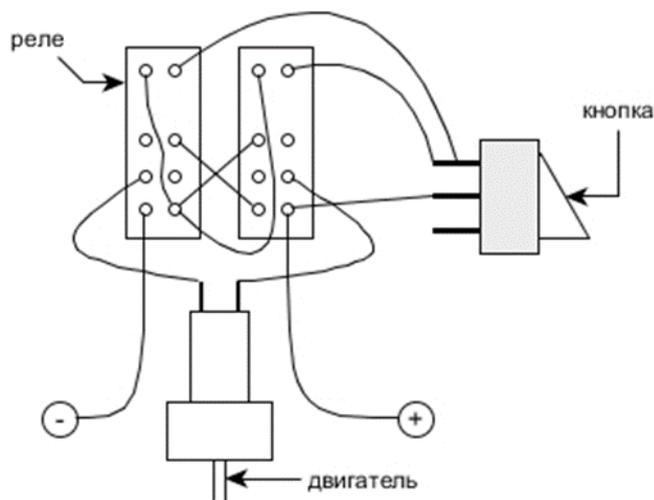


Рисунок 7 – схема осуществления реверсивного движения механизма наведения

5 Автоматическое наведение на цель

Для управления пусковым механизмом была выбрана плата управления Oktopod Studio [4], разработанная сербской командой участников соревнований Eurobot одного из предыдущих лет.

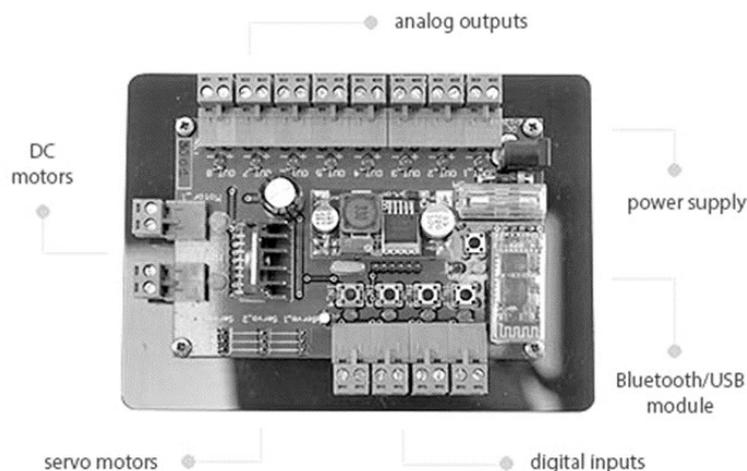


Рисунок 8 – Плата управления Oktopod Studio

Опишем алгоритм работы системы управления: датчик-камера распознает внешнюю среду и с помощью вычислителя передает информацию на плату. Oktopod Studio, получает координаты мишеней и управляет наведением пушки 1 и 2 этажей, в качестве меры безопасности, в случае срабатывания датчика-концевика (ограничителя перемещения труб) прекращает работу соответствующего двигателя. По готовности плата включает привод выстрела, и снаряд достигает желаемой цели. Подробный алгоритм решения задачи представлен на рисунке 9.

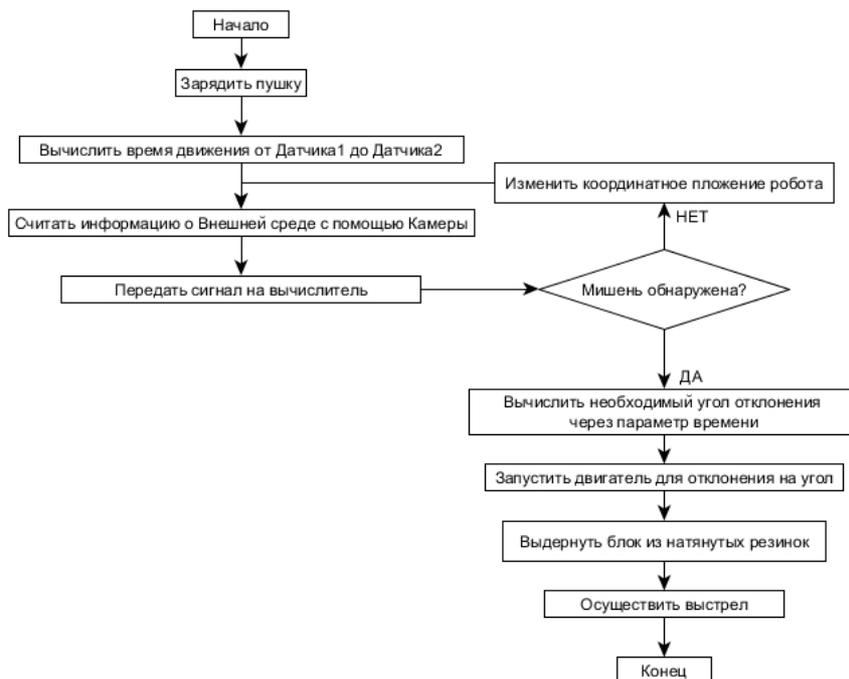


Рисунок 9 – Алгоритм работы модуля

С помощью программного обеспечения Oktopod Control - Software Instalation и Bluetooth-модуля осуществляется контроль и наладка пускового механизма. Достоинством данной платы является возможность управления мехатронным модулем в режиме онлайн (Рис. 10) или с помощью заранее заданной программы, которую в дальнейшем можно загрузить в память платы (Рис. 11).

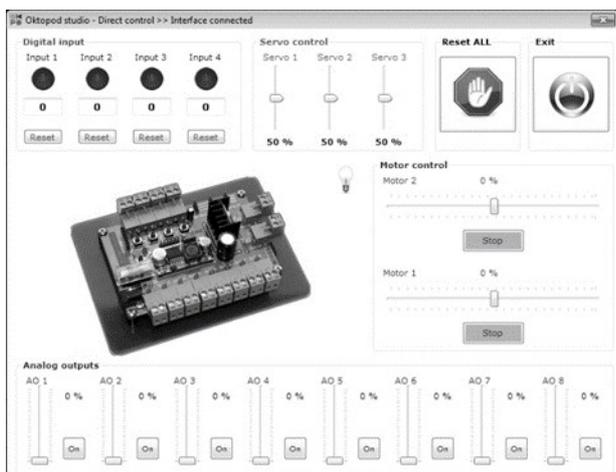


Рисунок 10 – Интерфейс онлайн управления (онлайн-отладка)

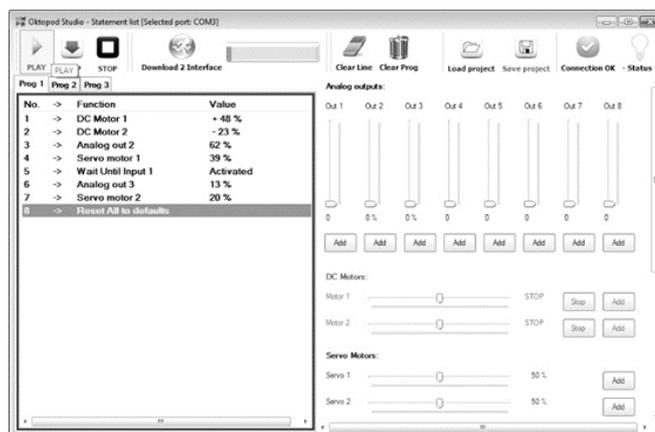


Рисунок 11 – Интерфейс программы управления (задание оффлайн-программы)

6 Система выстрела для снарядов

После того, как пушка направлена по требуемой траектории, необходимым остается только осуществить выстрел снарядов. Запуск мячиков происходит по 1 разу для каждого этажа пускового механизма. Перед каждым матчем соревнований участники должны самостоятельно зарядить пушку, натянув для каждого снаряда резинку на трубе. В момент выстрела подключенные к микроконтроллеру двигатели, не зависящие от двигателей наведения пушки, выдергивают держащие резинки чеки и снаряды вылетают с достаточной скоростью, для того, чтобы попасть в мишень. Выстрел осуществляется за 1 матч однократно, так как по условиям конкурса максимальное количество снарядов – 6 штук.

Заключение

Создание такого рода мехатронных модулей позволяет уменьшить время и трудоемкость осуществления проектов, что ведет к снижению стоимости изготавливаемого продукта, а значит, увеличивает его спрос и конкурентоспособность на рынке. На сегодняшний день важным направлением в робототехнике и проектировании является создание базы универсальных механизмов, которые можно использовать во многих проектах.

Технология создания мехатронных модулей, которая была освоена в процессе создания описываемой системы позволяет реализовать практически любой прототип для отработки идей конструирования успешного устройства, как для соревновательных целей Eurobot, так и для более серьезных коммерческих проектов. Технология отличается особым взглядом на командную работу и позволяет разработчикам более эффективно тратить время на разработку персональных мини-проектов, что позволяет в конечном итоге получить несколько хорошо отлаженных и описанных модулей для отработки конкретных задач мобильного робота.

В рамках данной работы был разработан пусковой механизм с системой автоматического наведения на цель. В проектировании использовались наиболее простые схемы и конструкции. Но задача разработки не ограничивается созданием одной модели, потому что необходимо исследовать различные стороны работы механизма, выявлять достоинства и недостатки, исправлять ошибки и повышать эффективность и надежность системы. Перспективна развития этой работы велика, чтобы в конце концов получить конечный вариант автономного пускового механизма с системой автоматического наведения на цель, удовлетворяющего конкурсу Eurobot и другим проектам и соревнованиям.

Литература

1. Intel® Edison Development Board. Электронный ресурс, Режим доступа: <http://www.intel.com/content/www/us/en/do-it-yourself/edison.html> – Проверено 31.01.2014.
2. International Students Robotic Contest Eurobot 2014. Электронный ресурс. Режим доступа: <http://www.eurobot.org/>, <http://www.eurobot-russia.org/home> — Проверено 31.01.2014.
3. Eurobot Junior 2014 Rules: «Prehistobot». Правила представлены на английском языке, 2013. — 36 с. Электронный ресурс. Режим доступа: http://www.eurobot.org/attachments/article/2/Eurobot2014_Rules_EN_Final_Version.pdf — Проверено 31.01.2014.
4. Open source development platform Oktopod Studio. Электронный ресурс. Режим доступа: <http://www.oktopodstudio.com/> — Проверено 31.01.2014.
5. Егоров О.Д., Подураев Ю.В. Мехатронные модули. Расчет и конструирование: Учебное пособие. — М.: МГТУ «СТАНКИН», 2004. — 360с.: ил.
6. Реле Song Chuan 793. Электронный носитель. Режим доступа: <http://lib.chipdip.ru/775/DOC000775255.pdf> — Проверено 31.01.2014.
7. Е.И. Забудский. Электрические машины. Ч. 4. Машины постоянного тока: Учебное пособие. — М.: МГАУ, кафедра Электроснабжение и Электрические машины, 2009. – 217 с.: ил.
8. А.Е.Аверьянихин, А.И.Власов, Л.В.Журавлева, Л.А.Зинченко, В.А.Соловьев ПРИМЕНЕНИЕ ПОЛОЖЕНИЙ ТЕХНИЧЕСКОГО РЕГЛАМЕНТА для оформления конструкторско-технологической документации при выполнении домашних заданий, курсовых работ и проектов / под ред. В.А.Шахнова. - М.: МГТУ им. Н.Э. Баумана, 2011. - 53 с.: ил.

ИССЛЕДОВАНИЕ ТЕХНОЛОГИИ СОЗДАНИЯ ЭЛЕКТРОННОГО МАКЕТА В NX ПОД УПРАВЛЕНИЕМ TEAMCENTER

Голиков С.А., Капитонов К.Д., Кошман А.А.
Научный руководитель: к.т.н., доцент, Маничев В.Б.
МГТУ им. Н.Э.Баумана, кафедра РК6, Москва, Россия

RESEARCH OF THE TECHNOLOGY OF CREATING ELECTRONIC MODEL IN NX CONTROLLED BY TEAMCENTER

Golikov S.A., Kapitonov K.D., Koshman A.A.
Supervisor: Ph.D., Associate professor, Manichev V.B.
BMSTU, RK6 Department, Moscow, Russia

Аннотация

В статье рассматривается методика создания электронного макета (ЭМИ) изделия с использованием подхода «сверху вниз». Кратко описывается структура ЭМИ и стадии его создания. Приводятся основные возможности PDM–системы Teamcenter. Описан принцип работы модуля «Технические Условия». В заключении представлены преимущества подхода «сверху вниз», а также ограничения сферы его применения.

Abstract

In this article the technique of creation of the Digital MockUp (DMU) of a product with approach use "from top to down" is considered. The structure of DMU and stage of its creation is briefly described. The main opportunities of PDM–system Teamcenter are given. The principle of operation of the Product and Manufacturing Information module is described. Advantages of approach "from top to down" are presented in the conclusion, and also restrictions of the domain of its application.

Введение

Большинство проектируемых изделий представляет собой сборочные единицы (далее, сборки). Сборка представляет собой некоторую иерархическую структуру, на разных уровнях которой находятся компоненты – под сборки или детали. Эта структура формируется в файле сборки (верхнего уровня). В настоящее время используются два основных подхода к проектированию.

В первом случае («снизу вверх») структура сборки получается автоматически при добавлении компонентов в сборку; во втором («сверху вниз») – разработка начинается с моделирования структуры сборки, за которым следует постепенное наполнение сборки входящими в нее компонентами.

Метод «снизу вверх» достаточно удобен при создании моделей сборок относительно простых изделий с небольшим количеством компонентов, когда все изменения в сборке отслеживаются одним-двумя конструкторами. Однако при увеличении числа компонентов и усложнении геометрии деталей возникают значительные трудности, связанные со сложностью согласования компонентов в сборке. В таком случае удобнее использовать подход «сверху вниз», постепенно детализируя форму и состав изделия.

1 Электронный макет изделия

Технология создания электронного макета использует методологию проектирования «сверху вниз». Проектирование начинается с создания базовой контрольной структуры (БКС), то есть модели, описывающей изделие на самом высоком уровне. БКС является компоновочной схемой изделия, в ней закладываются основные характеристики изделия, его внешний вид, конструкторские решения и многое другое. Благодаря существованию цифровой компоновки рабочее проектирование может осуществляться одновременно всеми инженерами, работающими над проектом. Цифровая компоновка является основой, от

которой отталкиваются конструкторы в ходе проектирования. Изделие проектируется изначально привязанным к БКС, следовательно, каждая проектируемая деталь позиционируется относительно системы координат изделия. В данном случае изделие практически избавлено от возможных нестыковок и пересечений компонентов в сборке.



Рисунок 1 - Структура личной контрольной структуры в NX

Каждый конструктор работает в личной контрольной структуре, в которую входят БКС, «рабочая» часть и «боевая» сборка.

2 Использование технологии WAVE

Технология электронного макета предполагает наличие ассоциативных связей между базовой контрольной структурой и компонентами сборки. Это реализуется с помощью технологии WAVE (What If Alternative Value Engineering). Основное назначение модуля WAVE заключается в создании и отслеживании связей между различными деталями, для чего создаются ассоциативные копии геометрии деталей. При создании связей между компонентами внесенные изменения автоматически распространяются на все связанные компоненты. WAVE позволяет копировать тела, поверхности, грани и другие объекты между деталями, где они могут быть использованы в качестве родительских объектов для геометрии.

Конструктор создаёт проектируемую деталь в «рабочей» части, отталкиваясь от базовой контрольной структуры и уже созданных деталей. С помощью технологии WAVE в проектируемую деталь добавляется необходимая обстановка, на основе которой строится геометрия рабочей детали. В рабочей детали содержится все дерево построения модели, и все последующие изменения геометрии осуществляются непосредственно в ней. После создания рабочей детали в «боевую» сборку добавляется ассоциативная WAVE-копия только твердого тела. Такой подход в значительной степени позволяет распараллелить процесс проектирования изделия. При правильном организованном использовании ассоциативных связей любые изменения геометрии и компоновки обрабатываются с высокой степенью автоматизации.

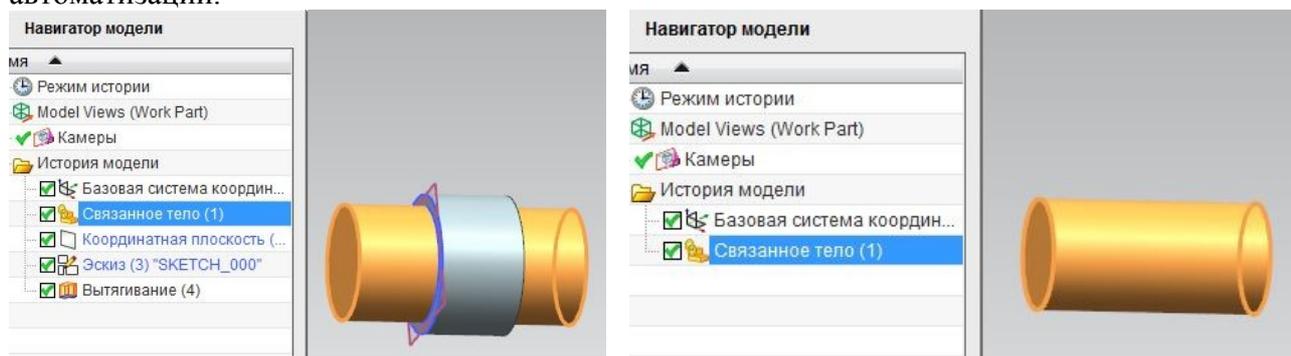


Рисунок 2 - Использование связей WAVE в «рабочей» и «боевой» деталях

При данном подходе создаётся один общий цифровой макет проектируемого изделия, который актуален в каждый текущий момент времени, то есть каждый элемент создается в логической или геометрической привязке к существующему на данный момент времени контексту проектируемого изделия.

3 Использование PDM системы Teamcenter для автоматизации процесса проектирования

Работа со сложными изделиями в едином информационном пространстве практически невозможна без использования PDM системы Teamcenter, позволяющей систематизировать большой объем данных, предоставить для конечного пользователя легкий доступ, поиск и обмен данными и обеспечить условия для эффективной совместной работы над изделием. В первую очередь PDM-система Teamcenter - это база данных, которая позволяет использовать одни и те же данные всеми участниками проекта, а механизм блокировки/разблокировки обеспечивает безопасность данных от внесения в них неправомерных изменений.

Каждый объект изделия (деталь, сборочная единица и др.) представлен в Teamcenter в виде Item – объекта базы данных, являющегося хранилищем всех связанных с ним данных. Каждый Item имеет уникальный идентификатор - Item ID и имя Item name (может быть не уникальным). Каждый Item имеет хотя бы одну свою модификацию (ревизию) ItemRevision. Она предназначена для контроля вносимых изменений в данные, которые хранятся в Item. Каждая ItemRevision имеет уникальный идентификатор. ItemRevision является потомком объекта Item. Важной составляющей Item изделия при создании электронного макета является ItemMaster / ItemRevisionMaster – мастер-форма (карточка), содержащая в себе описательную часть, атрибуты, описывающие тот или иной тип объекта Item и ItemRevision соответственно. Teamcenter позволяет создавать уникальную структуру мастер-формы под конкретные требования при проектировании того или иного изделия. Например, в карточке можно задавать габариты, материалы из созданной базы материалов, присваивать массу из модели и другие параметры.

4 Использование средств Product and Manufacturing Information (PMI)

Модуль NX «Технические условия» (PMI) является удобным инструментом для создания аннотаций для 3D-моделей изделия.

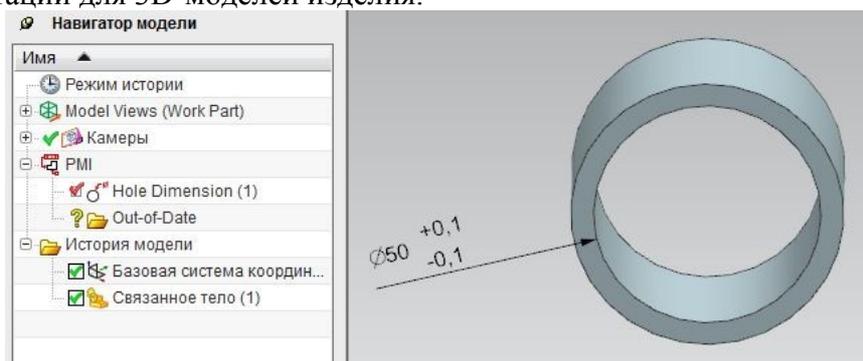


Рисунок 3 - Использование PMI

Данное приложение позволяет проставлять размеры, создавать замечания, отмечать шероховатость поверхностей и многое другое. Аннотированная модель является более наглядной и удобной при технологической подготовке производства, так как размеры и замечания, созданные в модуле «Технические условия» могут быть использованы при написании программ для станков ЧПУ.

Заключение

Использование технологии создания электронного макета изделия позволяет заметно сократить время конструкторской и технологической подготовки производства, обеспечить структурированный подход к проектированию, уменьшить объемы исправлений и автоматизировать процесс внесения изменений в геометрию и компоновку изделия. Такой подход позволяет использовать 3D-модель изделия, снабжённую PMI информацией, а также атрибутами из Teamcenter, в качестве подлинника при производстве.

Однако, применение данных технологий и сопутствующего программного обеспечения целесообразно на достаточно крупных предприятиях, имеющих большие объемы производства и обладающих значительными ресурсами для автоматизации процессов конструкторской и технологической подготовки производства. На малых же предприятиях затраты на установку и настройку программного обеспечения PDM нецелесообразны (расходы слишком велики по сравнению с доходами).

Литература

1. Гончаров П. С. и др. NX для конструктора-машиностроителя + CD. - М.: ДМК Пресс, 2010. - 504 с.
2. Данилов Ю., Артамонов И. Практическое использование NX. - М.: ДМК Пресс, 2011. – 332 с.
3. А.Г. Шилов Мы хотим стать инновационным, европейским предприятием по производству экскаваторов мечты // CAD/CAM/CAE Observer. - Выпуск №2(86). 2014.

РАЗРАБОТКА МОБИЛЬНОГО ИНТЕРАКТИВНОГО БРАУЗЕРА ДОПОЛНЕННОЙ РЕАЛЬНОСТИ

Амирханян А.Р.

Научный руководитель: к.т.н. Маничев В.Б.

МГТУ им. Н.Э.Баумана, кафедра РК-6, Москва, Россия

DEVELOPMENT OF MOBILE INTERACTIVE BROWSER FOR AUGMENTED REALITY

Amirkhanyan A.R.

Supervisor: Candidate of Engineering Science. Manichev V.B.

MSTU, Moscow, Russia

Аннотация

В статье рассматривается разработка мобильного интерактивного браузера дополненной реальности на платформе Android. Подробно исследованы особенности разработки программного комплекса, написанного с использованием свободно распространяемой библиотеки ARToolkit. Кратко представлена информация по современным областям применения дополненной реальности. В заключении представлены рекомендации по совершенствованию возможностей библиотеки ARToolkit.

Annotation

In this article the development of mobile interactive browser for augmented reality based on Android platform is considered. The specific cases of development software, based on opensource library ARToolkit are the main object of research. Information about modern spheres of use for augmented reality is briefly described. In conclusion recommendations for improvement of ARToolkit library are provided.

Введение

Дополненная реальность (ДР) совмещает реальный мир с виртуальными 3D-объектами в режиме реального времени. Технология дополненной реальности активно используется в различных областях жизнедеятельности человека. В медицине широко распространена компьютерная томография. Дополненная реальность позволяет отражать 3D-модели органов человека именно в том месте, в котором они расположены (рис. 1).



Рисунок 1 – Изображение ноги, совмещенной с 3D-моделью внутренних костей

В производстве дополненная реальность активно используется для проектирования производства. Технология используется на заводах компании Volkswagen [1] (рис. 2).



Рисунок 2 – Использование дополненной реальности для проектирования производства

В настоящее время разработка мобильных приложений, использующих дополненную реальность, ведется под платформы Android и iOS. Схема, иллюстрирующая типовой процесс работы подобных приложений, может быть представлена в следующем виде (рис. 3).

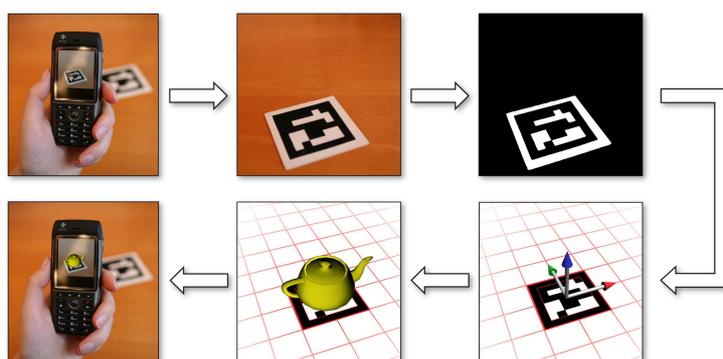


Рисунок 3 – Типовой процесс работы мобильных приложений, использующих дополненную реальность.

Задача, рассматриваемая в данной статье, состоит в разработке браузера дополненной реальности (ДР) на платформе Android и подтверждении того, что современные смартфоны на платформе Android могут быть успешно использованы для задач дополненной реальности, несмотря на их ограниченную вычислительную мощность. В статье приводится пример разработки на базе свободно распространяемой библиотеки ARToolkit. Данный программный комплекс может отражать 3D-модели на экране смартфонов на платформе Android с использованием маркеров дополненной реальности.

1 Информация о библиотеке ARToolkit

ARToolkit свободно распространяемая библиотека, разработанная Университетом Вашингтона (University of Washington), используемая во многих проектах, связанных с дополненной реальностью. Она написана на языке C и изначально была спроектирована под персональные компьютеры. Библиотека предлагает следующий функционал:

- Определение позиции/ориентации одиночной фотокамеры.

- Определение маркеров, использующих простые черные квадраты.
- Возможность использования любых шаблонов маркеров.
- Высокую скорость обработки в режиме реального времени объектов дополненной реальности.

Библиотеку можно скачать по интернет-адресу <http://www.hitl.washington.edu/artoolkit>. Она используется в разработанном браузере дополненной реальности.

2 Структура ARToolkit [2]

В разработанном браузере дополненной реальности в качестве базовой библиотеки используется ARToolkit. Задачи, стоящие перед браузером, состоят не только в определении расположения маркеров в пространстве и вычислении матриц трансляции 3D-объектов, но и в отображении 3D-объектов и их рендеринге с использованием OpenGL библиотеки. В таблице 1 показаны ключевые действия, которые производит браузер дополненной реальности. Шаги 2-5 повторяются в течении всего времени работы приложения. В ходе инициализации устройство вывода видеопотока инициализируется и ARToolkit считывает входные параметры. Шаги 2,3 и 4 в таблице соотносятся с вызовами библиотечных функций. На шаге 5 созданная матрица преобразований загружается в память и 3D-объект рисуется с использованием API OpenGL.

Таблица 1 – Ход работы приложения.

Инициализация	1. Инициализируется захват входящего видеопотока, чтение параметров шаблонов маркеров и видеокамеры.
Основной цикл	2. Захват видеопотока. 3. Определение маркеров и заданных шаблонов маркеров в видеопотоке. 4. Вычисление положения видеокамеры по отношению к распознанным шаблонам маркеров. 5. Отрисовка 3D-объектов на распознанных шаблонах маркеров.
Завершение работы	6. Закрытие входящего видеопотока.

Структура ARToolkit состоит из трех модулей.

1. Модуль дополненной реальности отвечает за вычисление расположения маркеров, калибровку камеры и сбор входных параметров.
2. Видео модуль отвечает за формирование видео фреймов на базе входящего видеопотока.
3. Графический модуль отвечает за формирование 3D-объектов. Модуль основан на базе библиотеки OpenGL.

Заключение

В данной статье была рассмотрена дополненная реальность в контексте мобильных устройств на платформе Android. Были представлены возможности библиотеки ARToolkit для разработки мобильного приложения с дополненной реальностью. Рассмотрены области применения дополненной реальности.

В качестве примера была проведена разработка мобильного интерактивного браузера дополненной реальности, способного отображать заданные пользователем 3D-объекты с учетом маркеров дополненной реальности и на основе входящего видеопотока.

В ходе запуска разработанного мобильного приложения на смартфоне Samsung Galaxy S3 с версией Android 4.0 была продемонстрирована удовлетворительная скорость вывода 3D-объектов в режиме реального времени (рис. 4, рис. 5).

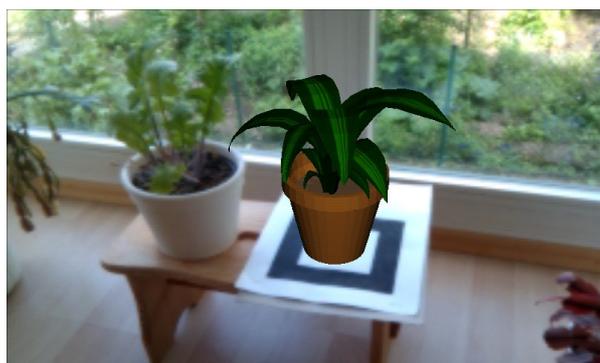


Рисунок 4 - Результат отображения 3D-объекта в режиме дополненной реальности



Рисунок 5 - Результат отображения 3D-объекта в режиме дополненной реальности

Литература

1. Katharina Pentenrieder, Christian Bade, Fabian Doil, and Peter Meier. Augmented reality-based factory planning - an application tailored to industrial needs. — Washington, DC, USA: Computer Science, 2007. — 19 с.
2. Daniel Wagner and Dieter Schmalstieg. Artoolkitplus for pose tracking on mobile devices. — Proceedings of 12th Computer Vision Winter Workshop, 2007. — 246 с.

МАТЕМАТИЧЕСКИЕ ТЕСТЫ ДЛЯ ЧИСЛЕННОГО МОДЕЛИРОВАНИЯ ВЫСОКОЧАСТОТНЫХ ЭЛЕКТРОННЫХ СХЕМ С БОЛЬШИМ РАЗБРОСОМ ПОСТОЯННЫХ ВРЕМЕНИ

Витюков Ф.А.

Научный руководитель: к.т.н, доцент Маничев В.Б.
Кафедра «САПР» (РК6) МГТУ им. Н.Э.Баумана, Москва, Россия

MATH TEST FOR NUMERICAL SIMULATION OF HIGH-FREQUENCY ELECTRONIC SCHEMES WITH LARGE TIME CONSTANTS DISPERSION

Vityukov F.A.

Scientific chief: PhD, associate professor Manichev V.B.
CAD/CAE Department (RK6) of BMSTU, Moscow, Russia

Аннотация

Предложен набор тестов для тестирования пакета Multisim версии 12.0 на возможность получения достоверного результата численного моделирования высокочастотных электронных схем с большим разбросом постоянных времени.

Abstract

Test cases set for Multisim 12.0 package is proposed to verify the possibility of obtaining reliable numerical simulation results for high-frequency electronic circuits with a large time constants dispersion.

Введение

Достоверность и точность численного моделирования электронных схем очень важны для автоматизированного проектирования электронных схем (EDA – Electronic Design Automation). Основным недостатком известных пакетов программ автоматизированного проектирования электронных схем (EDA пакетов), например, MicroCAP, Multisim, Symica и др., использующих решатель систем ОДУ SPICE, состоит в получении без диагностических сообщений неверного результата численного моделирования электронных схем при невысоких заданных требованиях к математической точности конечных результатов, т.е. невысокой заданной относительной точности ϵ_{ps} решения соответствующих систем ОДУ (обычно $\epsilon_{ps}=0.001$), математически моделирующих электронные схемы.

В данной работе рассматривается методика тестирования решателей систем ОДУ, реализованных в EDA пакетах, на математических тестовых задачах с известным асимптотическим или аналитическим решением. Рассмотрены тестовые системы ОДУ для численного моделирования высокочастотных электронных схем с большим разбросом постоянных времени, например, для численного моделирования тактовых кварцевых генераторов. Приведены результаты тестирования по данной методике EDA пакета Multisim версии 12.0.

В качестве примера одного из тестов приводится жесткая система ОДУ 2-го порядка с переменной степенью жесткости (MU – параметр жесткости) – осциллятор Ван дер Поля [4].

$$\begin{cases} dx_1 / dt = x_2 \\ dx_2 / dt = -x_1 + MU \cdot (1 - x_1^2) \cdot x_2 \\ x_1(0) = -1, x_2(0) = 1, \\ t \in [0, 4.2MU]. \end{cases}$$

Схемное представление системы ОДУ для ТЕСТа 1 реализуется с помощью источников тока, управляемых напряжением (ABM Current source) и единичных емкостей, показанных на рис.1.

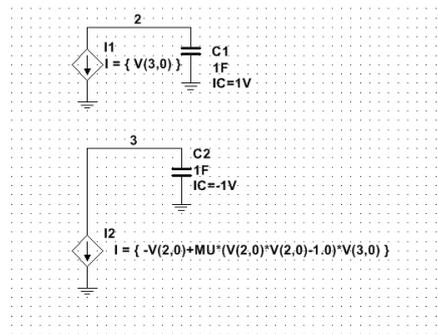


Рисунок 1 - Схемная модель системы ОДУ для осциллятора Ван дер Поля

Уже при $MU=10$ метод трапеций выдал не достоверный результат без диагностического сообщения об ошибках (рис. 2), при этом метод Гира выдал верный результат, показанный на рис. 3, но выдавал не достоверный результат для $MU>10^5$

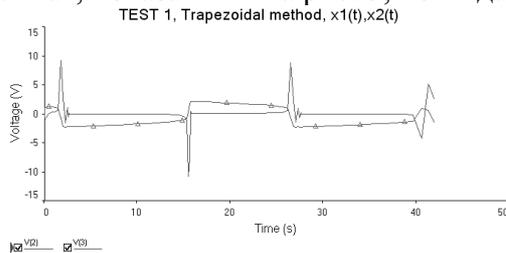


Рисунок 2 - Расчет теста 1 методом трапеций для $MU=10$

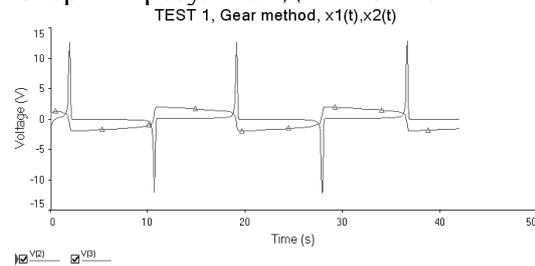


Рисунок 3 - Расчет теста 1 методом Гира для $MU=10$

Результаты тестирования EDA пакета Multisim версии 12.0 показали, что его не следует рекомендовать для моделирования высокочастотных схем с большим разбросом постоянных времени. Этот вывод совпадает с аналогичным выводом автора статьи [6].

Литература

- 1 O. Vityaz, V.Porra. Testing of Time Domain Simulators for Nonlinear Electronic Circuits. Helsinki University of Technology, Faculty of Electrical Engineering, Electronic Circuit Design Laboratory, Report 4, Finland, July 1988. 65 pages.
- 2 Alan C. Hindmarsh, Linda R. Petzold. Algorithms and Software for Ordinary Differential Equations and Differential Algebraic Equations. Numerical Mathematics Group Center for Computational Sciences@Engineering, UCRL-JC-116619, April 19 1994, 38 pages..
- 3 Guiyou Mao, L.R. Petzold. Efficient integration over discontinuities for differential-algebraic systems, Computers & Mathematics with Applications, Volume 43, Issues 1–2, January 2002, Pages 65-79.
- 4 Хайпер Э., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Жесткие задачи: Пер. с англ. -М.: Мир, 2001. - 498 с.
(Ernst Hairer, Gerhard Wanner. Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems. Springer-Verlag, 1996, 614 pages.)
- 5 Kundert K. The designer's guide to SPICE and Spectre. - Kluwer Academic Publisher. - 1995. - 371 с.
- 6 Макаренко В. Проблемы моделирования кварцевых генераторов в среде NI Multisim // Электронные компоненты и системы - 2010. - №7. - С. 43-49.

