МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДВНЗ «ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ (М. ВІННИЦЯ, УКРАЇНА)

ШТУТГАРТСЬКІЙ УНІВЕРСИТЕТ (М. ШТУТГАРТ, ФРН)

ТЕХНІЧНИЙ УНІВЕРСИТЕТ ГАМБУРГ-ХАРБУРГ (М. ГАМБУРГ, ФРН)

ІНСТИТУТ ПРОБЛЕМ МОДЕЛЮВАННЯ В ЕНЕРГЕТИЦІ ім. Г.Є.ПУХОВА НАН УКРАЇНИ

М А Т Е Р І А Л И

Сьомої міжнародної науково-технічної конференції

"МОДЕЛЮВАННЯ І КОМП'ЮТЕРНА ГРАФІКА"

18-24 вересня 2017 року
м. Покровськ, м. Київ

УДК 004.3+004.9+004.2+51.7+519.6+519.7

Публікується згідно з рішенням Вченої ради ДВНЗ «Донецький національний технічний університет» (протокол № 1 від 07.09.2017).

Збірник містить наукові статті співробітників ДонНТУ та інших навчальних і наукових закладів України, які взяли участь у роботі Сьомої міжнародної науково-технічної конференції "Моделювання і комп'ютерна графіка", що проводилась 18-24 вересня 2017 року у ДВНЗ «Донецький національний технічний університет» у м. Покровську. Публікації висвітлюють результати наукових досліджень і розробок в таких напрямках, як інформатика, чисельні методи, паралельні обчислення, програмування, розробка засобів обчислювальної техніки, дослідження комп'ютерних мереж, машинна графіка і обробка зображень, математичне моделювання в різних галузях. Матеріали збірника призначені для наукових співробітників, викладачів, інженерно-технічних працівників, аспірантів та студентів.

Видавець – Донецький національний технічний університет (ДонНТУ)

Конференція проводилась за підтримки:

- гранту Erasmus+ CBHE action 561728-EPP-1-2015-1-ES-EPPKA2-CBHE-JP «GameHub: University-enterprises cooperation in game industry in Ukraine»,
- гранту Tempus 2013–4587–001–001- 544010-TEMPUS-DE-TEMPUS-JPHES TATU – «Trainings in Automation Technologies for Ukraine»,
- компанії QuartSoft,
- компанії PHOENIX CONTACT.

Адреса редакції: 85300, м. Покровськ, пл. Шибанкова, 2, ДонНТУ.
Тел.: (06239) 2-09-38. E-mail: natalia.kostiukova@donntu.edu.ua

© Автори статей, 2017
© ДВНЗ «Донецький національний технічний університет», 2017

З М І С Т

Пленарні доповіді

Зорі С.А., Башков Є.О. **7**
ВИСОКОПРОДУКТИВНІ ЗАСОБИ РЕАЛІСТИЧНІЇ ПРОСТОРОВОЇ ВІЗУАЛІЗАЦІЇ ДЛЯ СИСТЕМ ТРЕХВИМІРОНОЇ КОМПЬЮТЕРОНОЇ ГРАФІКИ

Колесницький О.К. **20**
МОДЕЛЮВАННЯ СПАЙКІНГОВИХ НЕЙРОННИХ МЕРЕЖ

Романюк О. Н., Дудник О. О. **26**
РОЗРОБКА МЕТОДІВ ТЕКСТУРУВАННЯ ДЛЯ ЗАДАЧ ФОТОРЕАЛІСТИЧНОГО РЕНДЕРИНГУ

Секція 1.
Моделювання систем

Гуськова В.Г., Бідюк П.І. **34**
ПОБУДОВА СЦЕНАРІЇВ ІЗ ВИКОРИСТАННЯМ БАЙЄСІВСЬКИХ МЕТОДІВ

Дмитриева О.А., Половинка О.Л. **37**
ОСОБЕННОСТИ ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ ДЛЯ ВЫПОЛНЕНИЯ В РАСПРЕДЕЛЕННОЙ СРЕДЕ

Святний В.А., Гуськова Н.Г. **42**
ПРОЕКТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ МОДЕЛЮВАННЯ ВЗАЄМОДІЇ ПРОСТОРОВИХ СТРУКТУР

Дмитрієва О.А., Дараган Т.Г. **49**
ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ПРОГРАМНОЇ СИСТЕМИ ПАРАЛЕЛЬНОГО МОДЕЛЮВАННЯ ДИНАМІЧНИХ ОБ'ЄКТІВ

Бобилєва О. С., Дмитрієва О. А. **56**
ДОСЛІДЖЕННЯ ЕКСТРАПОЛЯЦІЙНОГО МЕТОДУ ДОРМАНА-ПРИНСА 5(4) З КЕРУВАННЯМ КРОКУ

Kovalov S. **59**
FUZZY CONTROL ALGORITHM SIMULATION

Лисенко Г. Л., Кошельна І. В., Бурмакіна О. В. **65**
МАТЕМАТИЧНА МОДЕЛЬ ПОГЛИНАННЯ ВИПРОМІНЮВАННЯ ДЛЯ ОПТОЕЛЕКТРОННОГО ТРАНСПАРАНТА З ДВОХВИЛЬОВИМ КЕРУВАННЯМ

Маслова Н.О., Половинка О.Л. **71**
ВИКОРИСТАННЯ МЕТОДІВ І АЛГОРИТМІВ ОБРОБКИ ВЕЛИКИХ ОБСЯГІВ ДАНИХ ПРИ РІШЕННІ ЗАДАЧ ФАРМАЦЕВТИЧНОЇ ЛОГІСТИКИ

Масюк А.Л., Мирошкин А.Н. **78**
МУЛЬТИПЛАТФОРМЕННЫЙ РЕДАКТОР ДВУМЕРНЫХ ГРАФОВ ОБЩЕГО НАЗНАЧЕНИЯ В КОНТЕКСТЕ МОДЕЛЬНОЙ ПРЕДМЕТНОЙ ОБЛАСТИ ГРАФОВ ШАХТНЫХ ВЕНТИЛЯЦИОННЫХ СЕТЕЙ
Фельдман Л.П., Назарова І.А., Гризадубова Я.А., Костін В.І. **84**
ЕФЕКТИВНІСТЬ ОБЧИСЛЕНЬ ПРИ ПАРАЛЕЛЬНОМУ РОЗВ'ЯЗАННІ ЗАДАЧІ КОШІ НА ГРАФІЧНИХ ПРОЦЕСОРАХ ЕКСТРАПОЛЯЦІЙНИМИ МЕТОДАМИ
Ануфрієв П. О. **92**
ПРОГРАМНА СИСТЕМА ДЛЯ ВИВЧЕННЯ ЛЕКСИКИ АНГЛІЙСЬКОЇ МОВИ
Федоров Є.Є., Патрушев В.О., Патрушева О.І. **95**
МЕТОД ПРОГНОЗУ ПРИБУТКУ ІНТЕРНЕТ МАГАЗИНУ НА ОСНОВІ ПОВНОЗВ'ЯЗНОЇ ДОВГОКОРОТКОЧАСНОЇ ПАМ'ЯТІ
Псьол В. О., Маслова Н. О. **108**
АЛГЕБРАЇЧНИЙ АНАЛІЗ ШИФРІВ, ОСНОВАНИХ НА КРИПТОГРАФІЧНИХ ПЕРЕТВОРЕННЯХ В АЛГЕБРІ ПОЛІВ ГАЛУА
Тищенко Г. В., Святний В. А., Бишенко Ю. А. **112**
СИСТЕМИ ЗБОРУ ДАНИХ З ВЕБ-РЕСУРСІВ

Секція 2.
High Performance Computing (HPC) and Parallel Simulation Technology (ParSimTech)

Resch M. **117**
BRINGING TOGETHER HIGH PERFORMANCE COMPUTING AND DATA ANALYTICS
Mangold M., Khlopov D., Lihong Feng L. **117**
NONLINEAR MODEL REDUCTION OF POPULATION BALANCE EQUATIONS BY POD-DEIM BASIS CONSTRUCTION
Погорілий С.Д., Бойко Ю.В., Сальніков А.О., Слюсар Є.А., Борецький О.Ф. **123**
ВИКОРИСТАННЯ СПЕЦІАЛІЗОВАНОГО ПРИКЛАДНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ RAINBOW У НАУКОВО-ДОСЛІДНИХ ГРІД-ПРОЕКТАХ

Секція 3.
Комп'ютерна графіка та візуалізація в ігрових додатках

Драченко М.Ю., Зорі С.А. **130**
ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ПАРАЛЕЛЬНОЇ РЕАЛІЗАЦІЇ АЛГОРИТМУ БІЛАТЕРАЛЬНОЇ ФІЛЬТРАЦІЇ ДЛЯ ВИРІШЕННЯ ЗАВДАНЬ РОЗПІЗНАВАННЯ ОБРАЗІВ НА ЗОБРАЖЕННЯХ ЗАСОБАМИ ТЕХНОЛОГІЇ CUDA
Глинська К.С. **136**
РЕАЛІЗАЦІЯ ШТУЧНОГО ІНТЕЛЕКТУ У СТРАТЕГІЧНИХ ІГРАХ НА ПРИКЛАДІ ІГРОВОГО ДОДАТКУ «ШАХИ»

Дзюбинский А.С., Дмитриева О.А. **142**
РАЗРАБОТКА МОДУЛЯ СИМУЛЯЦИИ ИГРОВЫХ ПРИЛОЖЕНИЙ В СРЕДЕ UNITY НА БАЗЕ ТЕХНОЛОГИИ KINECT

Кривенко О.М., Зори С.А. **145**
БАЗОВІ ПІДХОДИ ДО РЕАЛІЗАЦІЇ НАВІГАЦІЇ У ТРЬОХ ВИМІРНИХ ВІДЕОІГРАХ

Бишенко Ю. А., Костюкова Н. С., Тищенко Г. В. **148**
МЕТОДИ ПОБУДОВИ ВІРТУАЛЬНОГО ПРЕДМЕТНО-ПРОСТОРОВОГО РОЗВИВАЮЧОГО СЕРЕДОВИЩА

Лукін О. Ю., Костюкова Н. С. **152**
ЛОГІЧНА ГРА- ГОЛОВОЛОМКА З ВИКОРИСТАННЯМ ДЕДУКТИВНОГО МЕТОДУ ПОШУКУ У БАЗІ ДАНИХ

Костюкова Н.С., Башков Є.О., Матрьонін А.А. **157**
ІНТЕРАКТИВНА ВІЗУАЛІЗАЦІЯ ЗВ'ЯЗКІВ У РЕЛЯЦІЙНІЙ БАЗІ ДАНИХ

Костюкова Н.С., Назарова І.А., Роговець Є.В. **163**
АНАЛІЗ РИНКУ КОМП'ЮТЕРНИХ ІГОР ЗАСОБАМИ ОН-ЛАЙН СИСТЕМ АНАЛІТИКИ

Лефтеров М.М. **171**
АНАЛІЗ МОДЕЛІ ТА АЛГОРИТМІВ, ЩО МОЖУТЬ БУТИ ВИКОРИСТАНІ ДЛЯ ПОБУДОВИ ІГРОВОГО ФІЗИЧНОГО РУШІЯ ДЛЯ ТВЕРДИХ ЧАСТОК

Прядка Д. А., Маслова Н.О. **176**
ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ЗАСТОСУВАННЯ ПРОЦЕДУРНОЇ ГЕНЕРАЦІЇ В КОМП'ЮТЕРНИХ ІГРАХ

Nikolskyy A., Bilynskyy Y., Lazarev A., Krasilenko V. **178**
COMPUTERIZED SPERM QUALITY ANALYZER

Павлов С.В., Ганиш Н.В., Бурденюк І.І., Чернишова Т.А., Поплавська А.А. **183**
МЕТОД УЗАГАЛЬНЕНОГО КОНТУРНОГО ПРЕПАРУВАННЯ ДЛЯ ОБРОБЛЕННЯ БІОМЕДИЧНИХ ЗОБРАЖЕНЬ

Павлов С.В., Тимченко Л.І., Бурденюк І.І., Чернишова Т.А., Поплавська А.А **188**
ВИКОРИСТАННЯ МЕТОДУ УЗАГАЛЬНЕНОГО Q-ПЕРЕТВОРЕННЯ ЗОБРАЖЕННЯ В ЗАДАЧАХ ОБРОБЛЕННЯ НОРМАЛЬНОГО БІЛОГО ШУМУ

Романюк С.О, Безмертний Ю. О., Безсмертна Г. В. **193**
ТРИВИМІРНЕ МОДЕЛЮВАННЯ ДЛЯ ПЛАНУВАННЯ ТА ПРОВЕДЕННЯ ПЛАСТИЧНИХ ОПЕРАЦІЙ НА ОБЛИЧЧІ ЛЮДИНИ

Романюк О. Н, Романюк О. В., Войт Б. Л. **199**
АНАЛІЗ ГРАФІЧНОЇ АХІТЕКТУРИ NVIDIA PASCAL

Романюк С. О., Вяткин С.И. **204**
МЕТОД МНОГОПРОХОДНОГО РЕНДЕРИНГА ОБЪЕМНЫХ ДАННЫХ ДЛЯ МЕДИЦИНСКИХ ПРИЛОЖЕНИЙ

Romanyuk A. N., Vyatkin S. I. **208**
DEFORMABLE INTERACTIVE VOLUME-ORIENTED RENDERING BY PERTURBATION FUNCTIONS

Романюк О. Н., Мельник О.В., Стукач О.В. **215**
МОДЕЛЮВАННЯ ГЕКСАГОНАЛЬНОГО РАСТРА НА КВАДРАТНОМУ РАСТРІ
Тихонова О.А., Скрипник Т.В **219**
СТВОРЕННЯ ІГРОВОГО ДОДАТКУ ЗАСОБАМИ ГРАФІЧНОГО СЕРЕДОВИЩА UNITY 3D
Філатов І.А., Цололо С.О. **228**
ДОСЛІДЖЕННЯ МЕТОДУ ДЕКОНВОЛЮЦІЇ ВІНЕРА ПРИ РЕАЛІЗАЦІЇ НА МОБІЛЬНИХ ПЛАТФОРМАХ
Яровий А.А., Пасічник Д.Г., Василічишин Р.А. **234**
ІНТЕЛЕКТУАЛЬНА СИСТЕМА ЦИФРОВОЇ КОРЕКЦІЇ ТА ПІДВИЩЕННЯ ЯКОСТІ РАСТРОВИХ ЗОБРАЖЕНЬ
Шумейко А.К., Башков Є.О. **239**
ПРОГРАМНЕ ПІДКЛЮЧЕННЯ СЕНСОРУ KINECT ДО КОНСТРУКТОРУ GAME MAKER
Ермолов А.А., Костюкова Н.С. **244**
АНАЛИЗ МЕТОДОВ ПОИСКА НЕЧЕТКИХ ДУБЛИКАТОВ В БАЗЕ ДАННЫХ ИЗОБРАЖЕНИЙ

Секція 4.
Моделювання систем та об'єктів промислової автоматизації

Воропаєва В.Я. **249**
ВИКОРИСТАННЯ TATU SMART LAB У ПІДГОТОВЦІ ТА ПЕРЕПІДГОТОВЦІ ФАХІВЦІВ З ПРОМИСЛОВОЇ АВТОМАТИЗАЦІЇ
Дікова Ю.Л. **252**
СИСТЕМА ПРОГНОЗА СТАНУ ГІРНИЧОГО ОБЛАДНАННЯ НА БАЗІ МУЛЬТИАГЕНТНОГО ПІДХОДУ
Єшан Р.В., Поцепаєв В.В., Воропаєва В.Я. **258**
МОДЕЛЮВАННЯ ВОДОВІДЛИВНОЇ УСТАНОВКИ ЗАСОБАМИ МОБІЛЬНИХ ЛАБОРАТОРІЙ TATU
Поцепаєв В.В **263**
МАТЕМАТИЧНА МОДЕЛЬ НАВАНТАЖЕНЬ НА ВИКОНАВЧИХ ОРГАНАХ ВИДОБУВНОГО КОМБАЙНА
Тихонова О.А., Скрипник Т.В. **267**
СУЧАСНІ ЗАСОБИ НАВІГАЦІЇ У ВИРІШЕННІ КОМЕРЦІЙНИХ ЗАВДАНЬ, ПРОГРАМА "ROUTE"
Безбожний В.С., Цололо С.О. **274**
АВТОНОМНА СИСТЕМА КЕРУВАННЯ ЕЛЕКТРОННИМИ ЗАМКАМИ НА ОСНОВІ NFC
Новіков В.Г., Цололо С.О. **280**
ПЛАТФОРМА РОЗУМНОГО БУДИНКУ НА ОСНОВІ ARDUINO MEGA 2560

ПЛЕНАРНІ ДОПОВІДІ

УДК 004.92

ВЫСОКОПРОДУКТИВНЫЕ СРЕДСТВА РЕАЛИСТИЧНОЙ ПРОСТРАНСТВЕННОЙ ВИЗУАЛИЗАЦИИ ДЛЯ СИСТЕМ ТРЕХМЕРНОЙ КОМПЬЮТЕРНОЙ ГРАФИКИ

Е.А. Башков, С.А. Зори

Донецкий национальный технический университет,
г. Покровск, Украина

В работе дан краткий анализ разработанных авторами методов и средств для повышения производительности и реалистичности пространственной объемной визуализации в системах трехмерной компьютерной графики. Предложенные средства позволяют существенно повысить быстродействие, реалистичность и эффективность синтеза объемных изображений в системах 3D-пространственной визуализации.

Ключевые слова: 3D- объемная визуализация, синтез изображений, воксельная сцена, 3D- стерео, псевдо стерео.

1. ВВЕДЕНИЕ. АКТУАЛЬНОСТЬ

Современная компьютерная индустрия стремительно развивается, и в первую очередь в направлении эффективной машинной графической визуализации, обеспечивающей для человека самый высокий уровень информативности и наглядности важнейших аспектов происходящих явлений и процессов.

Сферы применения компьютерной графики сегодня чрезвычайно широки:

- системы трехмерного компьютерного моделирования в различных областях;
- системы медицинской диагностики и автоматизации выполнения операций;
- системы визуализации научных данных в научных областях;
- системы виртуальной и расширенной реальности (СВР);
- многоцелевые симуляторы и имитаторы;
- кинематография, индустрия развлечений, компьютерные игры;

- и многие другие.

Общий объем рынка систем компьютерной графики (СКГ) в 2016 году составил около140 млрд. долл., рынка СВР – 3,8 млрд. долларов.

Современное состояние СКГ и систем отображения информации (СОИ) характеризуется недостаточной степенью реализма синтезируемых изображений, особенно генерируемых в режимах реального времени динамических изображений. В связи с этим все основные направления исследований в этой области связаны с повышением производительности и реалистичности визуализации.

Последнее десятилетие задачи реалистичной визуализации и генерации статических и динамических реалистичных изображений переходят на новый качественный уровень – объемную визуализацию, которая позволяет повысить степень погружения в виртуальную реальность и качество визуальных ощущений от взаимодействия с ней. К сферам применений объемной визуализации относятся любые области научной, исследовательской и практической деятельности человека, потенциально требующие на сегодняшний день качественно нового уровня отображения результатов компьютерной симуляции и моделирования, а также качества визуальных ощущений от взаимодействия с ней.

Задача реалистичной пространственной визуализации на устройствах объемного отображения является сегодня относительно новой, активно обсуждаемой, имеющиеся результаты ее решения интенсивно внедряются в практическую область, разработке эффективных методов и средств объемной (пространственной) 3D-визуализации посвящено множество исследований во всем мире [1, 2]. Но, несмотря на это, единого эффективного подхода и обобщенных методик реализации качественного объемного отображения (как и соответствующих универсальных массовых технологий и высокопроизводительных устройств) на сегодняшний день пока не существует [1, 2, 3].

Организация систем реалистичной пространственной визуализации в общем случае предусматривает принципиально новую организацию вычислительного процесса, отличную от стандартного графического 3D конвейера, и его существенное усложнение за счет трудоемких процедур модельной 3D- генерации и трехмерной пространственной обработки, применения сложных методов синтеза и визуализации (таких, как трассировка лучей и пр.) для повышения реалистичности пространственного 3D- синтеза. Все это значительно ограничивает быстродействие формирования объемных изображений

графических сцен и, в свою очередь, делает практически невозможной реализацию динамических режимов реального времени.

Таким образом, проблема, связанная с несоответствием реалистичности и производительности формирования трехмерных объемных графических изображений требованиям большинства приложений и систем реалистичной пространственной 3D-визуализации реального времени, является неразрешенной и требует дальнейшего развития. Это актуализирует и порождает направленность теоретических и прикладных исследований в этой области - разработка эффективных и высокопроизводительных алгоритмических средств и параллельных архитектур программно-аппаратных комплексов для решения задач реалистичной пространственной 3D- визуализации.

1. 3D- СИНТЕЗ И ВИЗУАЛИЗАЦИЯ

Задача реалистичной пространственной визуализации и объемного отображения является общим случаем решения основной задачи трехмерной компьютерной графики, целью которой является создание механизма, обеспечивающего генерацию изображений, аналогичных тем, которые формируют зрительный аппарат и психофизиология человека, получая информацию из реального мира.

В настоящее время сложилась следующая укрупненная классификация синтеза и визуализации изображений, реализуемых СКГ/СОИ (рис. 1).



Рисунок 1 - Укрупненная классификация СКГ/СОИ

Трехмерная пространственная визуализация предполагает такое компьютерное преобразование модели сцены (пространственный 3D-синтез, рендеринг), которое позволит создать полностью трехмерный визуальный объемный образ (3D- изображение) в специфическом устройстве пространственного отображения (устройстве 3D-

визуализации). 3D- изображение затем некоторым способом отображается в экранном пространстве 3D- дисплея, и человек воспринимает его (возможно, с использованием дополнительных средств сопряжения) в объемном виде (рис. 2).



Рисунок 2 - Трехмерная пространственная визуализация

Выполнена классификация основных способов выполнения пространственного 3D- синтеза и визуализации (рис. 3) [1 - 3].

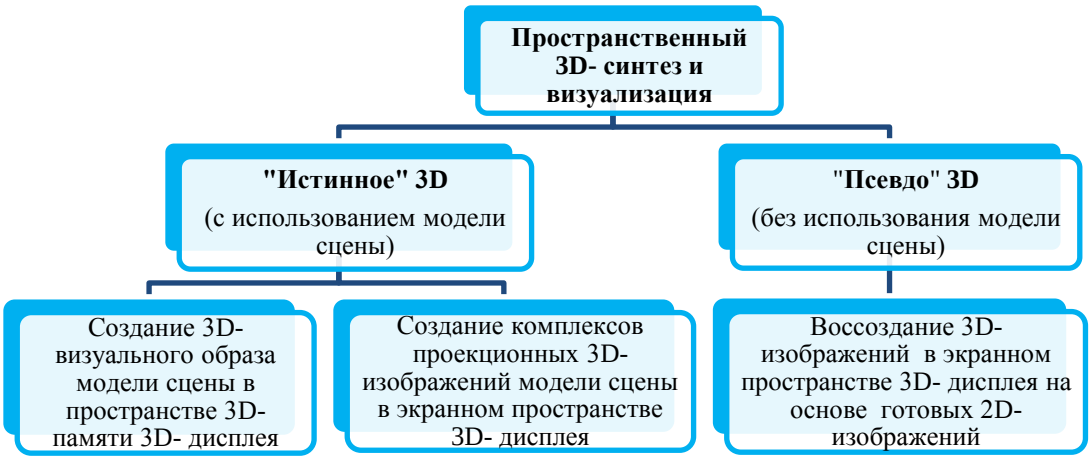


Рисунок 3 - Основные способы пространственной 3D- визуализации

Определены основные способы организации систем 3D визуализации - системы стереоскопической 3D- визуализации и системы пространственной объемной 3D- визуализации.

Принципиальное отличие систем пространственной объемной 3D- визуализации состоит в том, что в них фактически не требуется

выполнение проецирования сцены на плоскость экрана и выполнения традиционного набора процедур графического конвейера, а расчет сцены состоит по сути в создании дискретизированных объемно-пространственных моделей ее трехмерных объектов - визуальных объемных образов объектов сцены, которые в дальнейшем визуализируются на специализированных устройствах пространственного отображения - 3D- объемных дисплеях.

Системы объемной пространственной 3D- визуализации и дисплеи не позволяют пока получать полноценные и высококачественные 3D-изображения, имеют высокую стоимость и ограниченный круг пользователей. Для них характерно отсутствие стандартизации представления 3D информации, типовых 3D графических примитивов и алгоритмов их генерации.

Таким образом, решение задачи повышения производительности и качества формирования трехмерных объемных изображений является важной проблемой, решение которой порождает прикладные исследования по созданию эффективных методов и параллельных архитектур вычислительных комплексов для реализации систем реалистической пространственной визуализации с использованием технологий объемного отображения.

2. ПОВЫШЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ 3D- СОИ

Сформулированы основные направления повышения производительности реалистичного компьютерного синтеза трехмерных изображений для основных типов систем 3D-визуализации. Основными ключевыми моментами здесь являются производительный традиционный и объемный рейтрейсинг на основе ускоряющих техник, производительная генерация вокселизированных 3D- моделей мира и 3D- графических примитивов, быстрого синтеза на основе готовых изображений, а также возможности их распараллеливания и отображения на параллельные архитектуры вычислительных систем аппаратной поддержки.

2.1 Повышение производительности систем объемной пространственной визуализации

Наиболее перспективными и малоисследованными на сегодня являются системы, основанные на способах объемной пространственной визуализации, и, в первую очередь, базирующиеся на воксельном представлении трехмерной объемной сцены.

Для решения задачи эффективного синтеза изображений в

системах объемной пространственной визуализации авторами предложена обобщенная концепция представления дискретного объемного трехмерного мира и синтеза 3D- объемных изображений, даны ее основные определения, характеристики и особенности [2, 3, 13]. Это обеспечивает концептуальную категоризацию, смысловую интерпретацию и структурный базис организации таких систем.

Главная особенность предложенного подхода – фактически, при наличии готовых пространственно-дискретизированных (воксельных) моделей объектов мира (и других характеристик дискретного мира) построение изображения можно свести к определению видимых наблюдателю вокселей моделей объектов, попавших в видовой пространственной объем объемного дисплея, с дальнейшим учетом в их характеристиках параметров освещения и других визуальных свойств внутреннего окружающего пространства мира [2, 3, 13].

В рамках предложенной концепции разработан обобщенный метод синтеза объемного изображения воксельного дискретного мира для систем 3D- СОИ [2, 3, 13]. Предложенный подход сочетания пространственно-дискретного представления мира (сцены) и пространственно- дискретизированной трассировки лучей позволяет достаточно просто алгоритмически выполнить построение объемного изображения, фактически одновременно решая сразу множество задач–стадий графического 3D-конвейера – сценарного процессинга, модельных преобразований, рендеринга (включая удаление невидимых поверхностей и освещение), и собственно визуализации.

На основе разработанных концепции и обобщенного метода предложена обобщенная архитектура синтеза объемного изображения в воксельной 3D- СОИ (рис. 4). Сформулированы две основные задачи для эффективной реализации предложенных средств - эффективная генерация воксельных представлений объектов мира и эффективное выполнение дискретной трассировки лучей.

Предложен обобщенный метод генерации графических пространственных примитивов для объемных 3D- дисплеев [1 - 3, 12], на основе которого разработаны методы и алгоритмы генерации воксельного представления базовых графических пространственных 3D- примитивов (рис. 5) - пространственного сферического треугольника, отсека пространственного параболоида, плоского пространственного треугольника, произвольной пространственной кривой, отрезка прямой, выполнена оценка ошибки и временных затрат предложенных методов воксельного разложения 3D графических примитивов. Проведены эксперименты по реализации предложенных алгоритмов генерации графических пространственных

примитивов для объемных 3D дисплеев, подтвердившие их работоспособность и эффективность. Предложенные алгоритмы позволяют более чем в 3 раза увеличить производительность генерации.

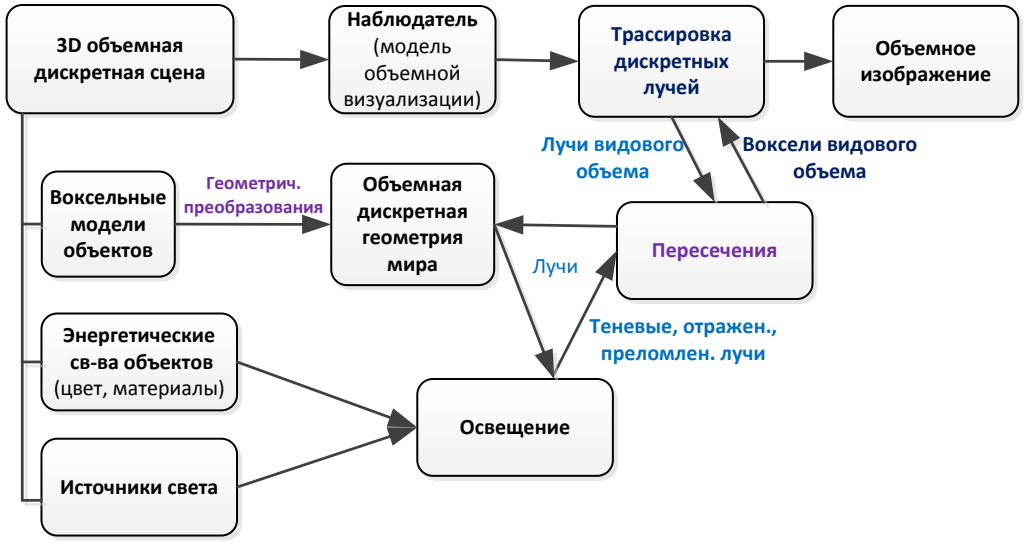
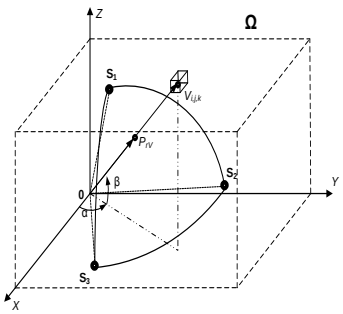
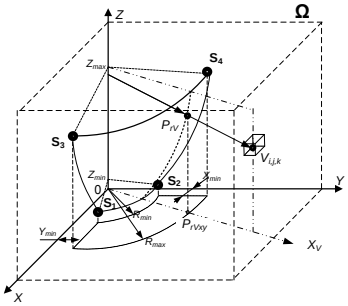


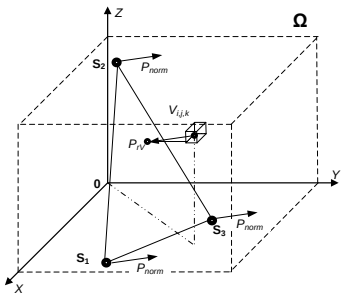
Рисунок 4 - Обобщенная архитектура синтеза в воксельной 3D- СОИ



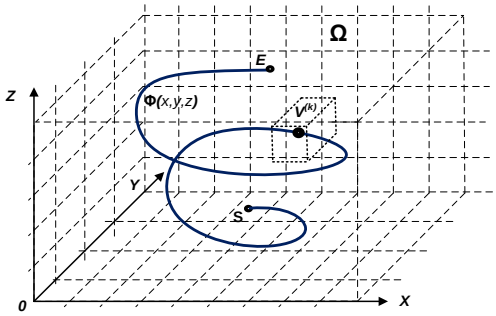
а) пространственный сферический треугольник



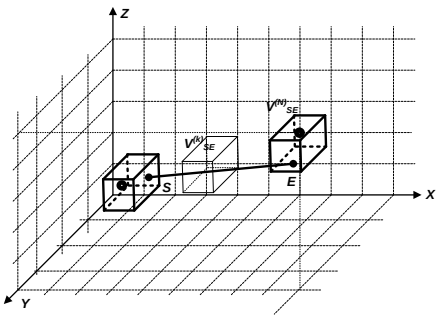
б) пространственный эллиптический параболоид



в) пространственный плоский треугольник



г) пространственная кривая



д) пространственный отрезок

Рисунок 5 - К воксельному представлению пространственных 3D-примитивов

Показано, что для решения задач 3D- визуализации в реальном времени необходимо применение средств высокопроизводительной параллельной аппаратной поддержки. Предложена обобщенная архитектура параллельной вычислительной системы для аппаратной поддержки объемной пространственной 3D- визуализации (рис. 6).

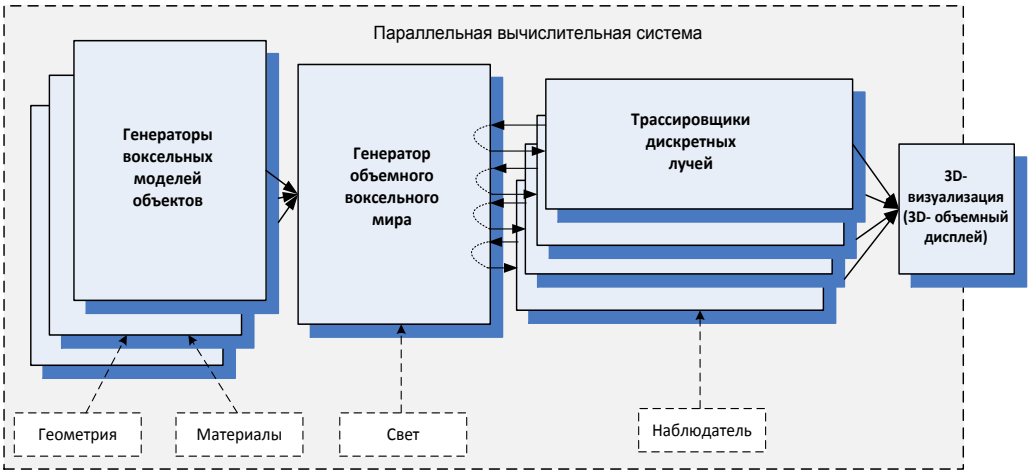


Рисунок 6 - Архитектура воксельной 3D- СОИ

Определено основное направление реализации аппаратного ускорения систем 3D- СОИ - использование аппаратного обеспечения параллельных графических процессоров (GPU) как наиболее распространенного, не дорогого и доступного варианта реализации, де-факто ставшего самым доступным подходом к реализации параллельных многопоточных вычислений. Для организации ускорения синтеза динамических изображений можно использовать временную и пространственную декомпозицию процесса обработки.

На базе архитектуры GPU разработана организация вычислительного процесса генерации 3D- объемных графических примитивов на основе разработанных алгоритмов их воксельной генерации [1, 2, 3, 12]. Выполнено экспериментальное исследование предложенных средств. Достигнутое ускорение при использовании предложенных методов на тестовом стенде составило более 17 раз.

2.2 Повышение производительности систем пространственной 3D- стерео визуализации

Предложена обобщенная архитектура высокопроизводительного реалистичного пространственного 3D- стерео синтеза и визуализации изображений (рис. 7) [1, 2, 6, 7, 9, 11].

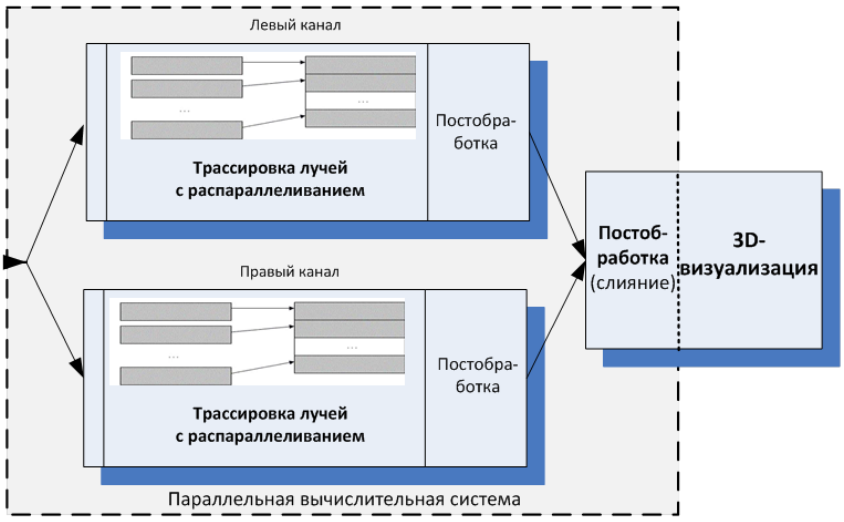


Рисунок 7 - Архитектура системы 3D- стерео синтеза и визуализации

Основные идеи повышения производительности и реалистичности – использование трассировки лучей как метода синтеза реалистичных изображений, его ускорения и распараллеливания, распараллеливание обработки в левом/правом каналах стерео синтеза ввиду ее полной идентичности, использование средств высокопроизводительной аппаратной поддержки.

Поиск пересечений луча с геометрическими объектами всегда является узким местом в трассировке лучей (около 75% времени формирования изображения). Для ускорения решения этой задачи используют ускоряющие техники трассировки лучей (УТ). Выполнена классификация УТ, дан анализ особенностей и характеристики основных используемых УТ [1, 4, 6, 8, 10, 11]. Для повышения эффективности выполнения рейтрейсинга предложено в качестве УТ использовать структуру BVH с AABB из-за простоты и скорости реализации, а также разработана ее модификация для ускорения реализации при многопоточных- и GPU- вычислениях. Использование модифицированного алгоритма AABB для тестовых сцен позволило сократить время поиска пересечений луча с объектами сцен от 4,5 до 24 раз по сравнению с вариантами без УТ [1, 4, 6, 10, 11]. Однако исследования показали, что реализация системы рейтрейсинга с выполненной модификацией в однопоточной реализации не позволяет синтезировать изображения в реальном времени и необходима параллельная реализация разработанной алгоритмической базы.

Выполнено отображение предложенной общей архитектуры 3D-стерео визуализации ускоренным методом трассировки лучей AABB на GPU и отображение обобщенной процедуры 3D- стерео синтеза на

GPU/GPC, выполнено исследование ее эффективности при реализации на GPU по сравнению с однопоточной и многопоточной реализациями на CPU тестовых стендов [7, 9, 11]. Результаты показали, что:

- модифицированный алгоритм дает прирост производительности по отношению к базовому варианту от 27% до 32% при его реализации на GPU тестового стенда без оптимизации реализации на конкретной архитектуре GPU;
- общее среднее достигнутое ускорение при многопоточной реализации на тестовом GPU (16 ядер) составляет около 7,5 раз и 1,55 раз при многопоточной реализации на тестовом CPU (4 ядра) по сравнению с однопоточной реализацией;
- показано, что полученные оценки могут быть существенно улучшены при реализации на новых архитектурах GPU с большим количеством вычислительных ядер и мультипроцессоров, а также проведении оптимизации конфигурации вычислительной CUDA-сети.

2.3 Повышение производительности систем пространственной 3D- псевдо стерео визуализации

Предложен обобщенный метод получения 3D- стереокадра из 2D изображения для организации реалистичной 3D- псевдо стерео визуализации графического и видео контента. Разработаны детальные алгоритмы этапов преобразования 2D- изображений в 3D- псевдо стерео на основе современных подходов к процессу и их частичного упрощения для ускорения генерации [2, 5, 14]. Выполнено экспериментальное исследование преобразования 2D- изображений в 3D- псевдо стерео с визуализацией на стандартных устройствах 3D- стерео отображения (анаморфный SBS 3D- стерео формат) и 2D- мониторе (анаглифный 3D- стерео формат), показавшее работоспособность предложенного метода, алгоритмов и упрощений.

Для повышения производительности системы 3D- псевдо стерео визуализации графического и видео контента предложена обобщенная архитектура параллельной вычислительной системы (рис. 8).

Разработана организация средств параллельной аппаратной поддержки для предложенного алгоритмического обеспечения 3D- псевдо стерео визуализации на архитектуре GPU- CUDA. Выполнена CUDA- реализация этапов алгоритма и исследование эффективности реализации предложенной алгоритмической базы синтеза 3D- псевдо стерео на GPU тестовой системы. Результаты показывают следующее:

- достигнутое ускорение вычислительных процедур 3D- псевдо-стерео синтеза для анаглифного и анаморфного форматов 3D-

стереокадра без выполнения оптимизационных процедур в среднем составляют 11 и 54 раза для тестовых GPU;

- возможна дополнительная оптимизация реализации на CUDA приведенных реализаций этапов (вариация размеров блоков сети CUDA дает среднее дополнительное ускорение в 1,25 раза), что актуализирует проведение правильного выбора конфигурации вычислительной сети и других оптимизационных процедур;

- выполнен теоретический прогноз роста производительности от количества ядер мультипроцессора GPU (который, однако, не учитывает такие определяющие характеристики GPU, как тип микроархитектуры, частоту работы GPU и памяти и пр.), показано, что реализация системы на современных GPU (более 2000 ядер) делает возможным реализацию процесса в реальном времени.

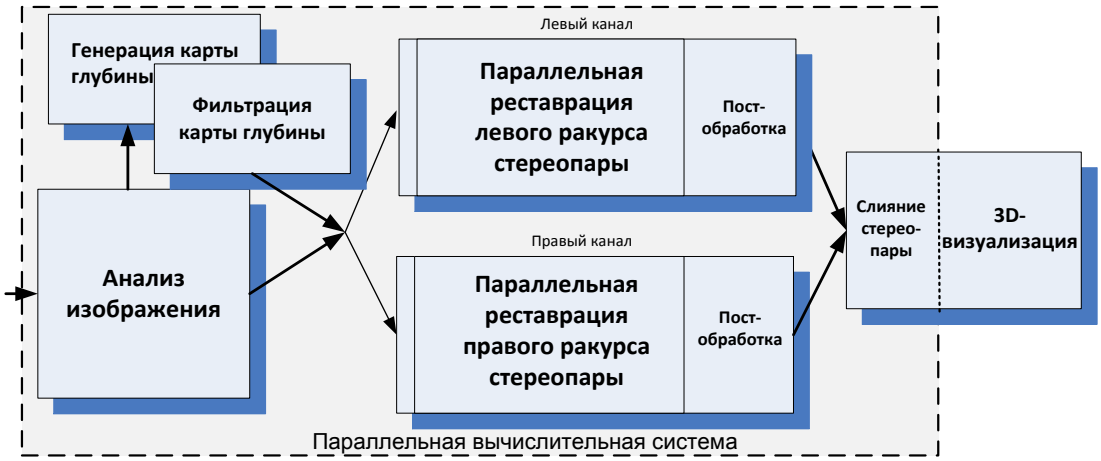


Рисунок 8 - Архитектура системы псевдо 3D- стерео синтеза

3. ВЫВОДЫ

В работе дан краткий анализ разработанных авторами теоретических основ и практических подходов к их реализации для решения актуальной научной проблемы повышения производительности и реалистичности пространственной объемной визуализации систем трехмерной компьютерной графики. Разработанные методы и средства их реализации позволяют существенно повысить быстродействие, реалистичность и эффективность систем 3D- пространственной визуализации. На основе предложенных положений разработаны работоспособные высокопроизводительные практические реализации для синтеза изображений и 3D- пространственной визуализации в различных прикладных областях.

ЛИТЕРАТУРА

[1] **Е.А.Башков, С.А.Зори** Реалистичная пространственная визуализация с использованием технологий объемного отображения: Монография. - Донецк, ГВУЗ "ДонНТУ", 2014. - 150 с.

[2] **Башков Е.А., Зори С.А.** Системы пространственной визуализации окружающей обстановки // Науковий вісник Донецького національного технічного університету – Красноармійськ, ДонНТУ, Міністерство освіти та науки України, 2016. - №1 (1). - С. 20-45.

[3] **Башков Е.А., Зори С.А.** Организация синтеза объемных изображений дискретных воксельных сцен в системах объемной пространственной визуализации // Наукові праці Донецького національного технічного університету. Серія: "Інформатика, кібернетика та обчислювальна техніка". - №1 (22), 2016. - С. 5-12.

[4] **С.А. Зори, И.А. Запорожченко, М.А. Григорьев** Анализ алгоритмов трассировки лучей для реалистичной визуализации трехмерных сцен и способов уменьшения их вычислительной сложности // Цифровая обработка сигналов и изображений. — 2013. — С. 352–357.

[5] **Зори С.А., Звягинцев И.А.** Организация псевдо 3D-визуализации графического и видео контента // Наукові праці Донецького національного технічного університету. Серія: "Інформатика, кібернетика та обчислювальна техніка". - №1 (22), 2016. - С. 13-22.

[6] **Зори С.А.** Объемная визуализация алгоритмом трассировки лучей с использованием двухуровневой иерархии ограничивающих объемов и AABB // Наукові праці Донецького національного технічного університету. Серія: **—** Інформатика, кібернетика та обчислювальна техніка. №2 (21)' 2015.- С. 5-10.

[7] **Зори С.А.** GPU- реализация параллельной вычислительной системы 3D- стерео визуализации с использованием метода трассировки лучей // Системи обробки інформації: Збірник наукових праць. – Х.: Харківський університет Повітряних Сил ім. Івана Кожедуба Міністерства Оборони України, 2016. – Вип. 6 (143). – С. 201-204.

[8] **Зори С.А.** К вопросу определения временной сложности алгоритмов трассировки лучей // Міжнародний науково-технічний журнал "Вимірювальна та обчислювальна техніка в технологічних процесах". - Хмельницький: Хмельницький національний університет. - № 2(55), 2016. - С. 114 – 118.

[9] **Зори С.А.** Синтез трехмерных стерео изображений в параллельной вычислительной системе 3D- стерео визуализации // Наукові праці Донецького національного технічного університету. Серія: — Інформатика, кібернетика та обчислювальна техніка. №1 (24)' 2017.- С. 44-52.

[10] **S. A. Zori, P. A. Porfirov** Productivity Increasing of Realistic Ray Tracing Stereo- Image Synthesis //Journal of Qafqaz University, Mathematics and Computer Science ISSN 1302-6763, Vol. 3, No 1, 2015.- P. 30- 38.

[11] **Zori S.A., Al-Oraiqat Anas M.** 3D- Visualization by Raytracing Image Synthesis on GPU // International Journal of Engineering Science and Technology (IJEST). - Vol. 8, № 06, Jun. 2016. – P. 97 – 104.

[12] **Bashkov E.A., Zori S.A., Al-Oraiqat Anas M.** Resolution Improvement of the Common Method for Presentating Arbitrary Space Curves Voxel // International Journal of Electronics Communication and Computer Engineering (IJECCE). - Vol. 7, Issue 6, Dec. 2016. – P. 326 – 330.

[13] **Bashkov E.A., Zori S.A., Al-Oraiqat Anas M., Aladdein M. Amro** Generalized 3D Voxel Image Synthesis Architecture for Volumetric Spatial Visualization // International Journal of Information Technology (IIJIT). - Vol. 5, Issue 1, Jan. 2017. – P. 1 – 9.

[14] **Zori S.A., Al-Oraiqat Anas M., Aladdein M. Amro** Conceptual and algorithmic development of Pseudo 3D Graphics and Video Content Visualization // International Journal of Information Technology (IIJIT). - Vol. 5, Issue 1, Jan. 2017. – P. 17 – 28.

Отримано 21.06.2017

УДК 004.93

МОДЕЛЮВАННЯ СПАЙКІНГОВИХ НЕЙРОННИХ МЕРЕЖ

О.К. Колесницький, к.т.н., доцент
Вінницький національний технічний університет

Наведено результати комп'ютерного моделювання спайкінгової нейронної мережі в задачі розпізнавання імпульсних послідовностей з метою визначення кращого алгоритму навчання мережі. Із двох досліджених алгоритмів: зворотного розповсюдження та лінійної класифікації кращим за критерієм достовірності та значенням помилки розпізнавання виявився алгоритм лінійної класифікації.

ВСТУП

Останнім часом все більшої популярності набувають так звані спайкінгові нейронні мережі [1,2]. Вони, на відміну від традиційних нейронних мереж, в яких інформація кодується статичними потенційними сигналами (бінарними або аналоговими), використовують імпульсне кодування інформації, яке притаманне біологічним нейронним мережам, а тому є більш перспективним для моделювання штучних інтелектуальних систем [3].

Спайкінгові нейронні мережі мають перед традиційними нейронними мережами такі переваги [1,3]:

1) розпізнавання динамічних образів без попереднього їх перетворення у вектор ознак;
2) багатозадачність;
3) розпізнавання з передбаченням;
4) простота процедури навчання.

Перша перевага робить їх незамінними при розпізнаванні динамічних сигналів. Задача розпізнавання динамічних сигналів являє собою досить актуальну проблему у різних галузях науки й техніки. Прикладами таких сигналів у медицині є кардіограма, у геології - луна від землетрусу, в астрономії - графіки сонячної активності, сигнали від далеких галактик, прийняті радіотелескопом, в економіці – зміни рівня безробіття чи процентних ставок, або динаміка змін курсу валют та акцій.

Людина сприймає образи через органи чуттів (очі, вуха, ніс і т.д.), які являють собою багатокомпонентні сенсори (рецептори). Ці багатокомпонентні сенсори видають багатоканальну інформацію у вигляді сукупності імпульсних послідовностей, або сигналів. Для Центральної нервової системи людини об'єктом є сукупність сигналів, що генеруються рецепторами, які згодом проходять складну обробку.

Тому при побудові нейроморфних комп'ютерних систем нового покоління (систем штучного інтелекту) [4,5,6] велике значення має вирішення проблеми розпізнавання багатокомпонентних сигналів чи, іншими словами, багатовимірних імпульсних послідовностей.

ПОСТАНОВКА ЗАДАЧІ

Отже, потрібно промоделювати роботу спайкінгової нейронної мережі при розв'язанні задачі розпізнавання багатовимірних імпульсних послідовностей. Для досягнення достовірного та безпомилкового розпізнавання імпульсних послідовностей потрібно дослідити роботу мережі після її навчання за різними алгоритмами навчання та обрати той з них, що приводить до більш достовірних результатів та меншої помилки розпізнавання.

Реалізація процесу моделювання

Для комп'ютерного моделювання роботи системи, що виконує розпізнавання багатовимірних імпульсних послідовностей, було обрано програмне середовище MATLAB, що містить додатковий модуль CSIM [7] для реалізації спайкінгових нейронних елементів [8] та мереж.

Робота системи розпізнавання багатовимірних імпульсних послідовностей відбувається наступним чином:

1 Ініціалізація шаблонів імпульсних послідовностей. Для конкретизації процесу моделювання було прийнято тривимірні імпульсні послідовності. Кожна імпульсна послідовність задається рядом моментів часу виникнення імпульсів протягом періоду 1с. У систему заноситься інформація про класи імпульсних послідовностей (шаблони), які в подальшому будуть розпізнаватись. Кількість шаблонів – 10.
2 Генерація вхідних імпульсних послідовностей. Формування імпульсних послідовностей для тестування та навчання із шаблонів шляхом внесення шуму - зсуву (випадкова величина, розподілена за нормальним законом з нульовим математичним сподіванням та дисперсією 4мс). Об'єм навчальної вибірки складає 800 імпульсних послідовностей, тестової – 200.
3 Створення та навчання спайкінгової нейронної мережі. Нейронна мережа має наступну структуру:
   - кількість нейронів мережі у вхідному шарі, дорівнює числу вхідних послідовностей, тобто 3;
   - кількість нейронів внутрішнього шару 150. Виходячи із даних нейрофізіологічних досліджень [1], в спайкінговій нейронній

мережі має міститися 20% гальмуючих нейронів. Ці нейрони обираються довільно.;

- кількість нейронів у вихідному шарі – 12.

З метою подальшого виявлення кращого за критерієм достовірності та значенням помилки розпізнавання алгоритму навчання мережі було використано алгоритми навчання зворотного розповсюдження та лінійної класифікації [7]. Цільовий вихід має вигляд номера шаблона, з якого була сформована імпульсна послідовність навчальної вибірки.

4 Перевірка результатів розпізнавання тестових імпульсних послідовностей. Фіксація значення виходу порогового елементу, який обробляє значення вихідного шару нейронів мережі після подання на вхід мережі тестових імпульсних послідовностей, та порівняння його з цільовим виходом порогового елементу.

В результаті роботи розробленої системи отримане зображення спайкінгової нейронної мережі, що моделюється (рис 1), яке містить вхідний, внутрішній та вихідний шари нейронів та відображає деякі зв'язки між нейронами.

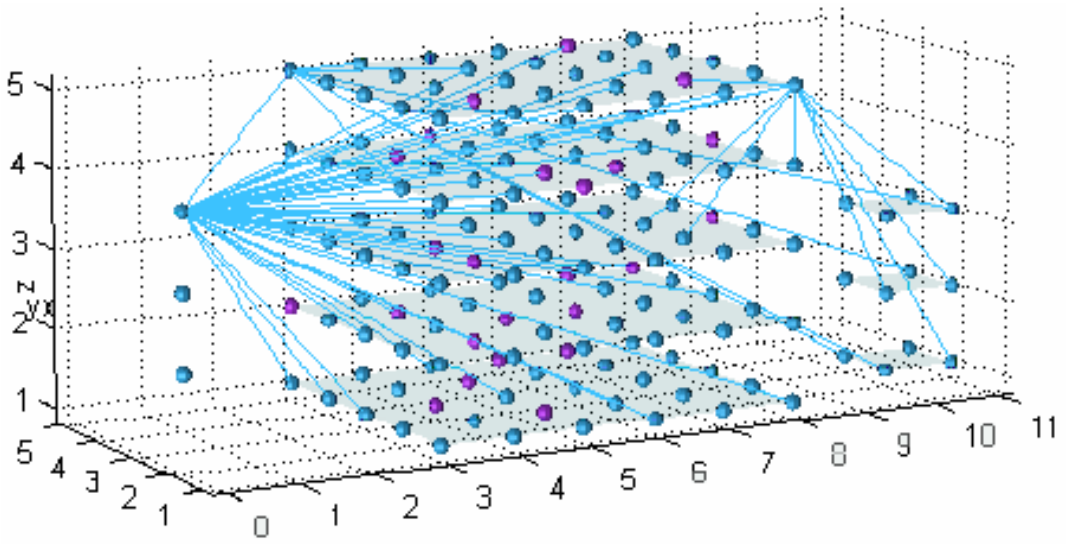


Рисунок 1 - Зображення спайкінгової нейронної мережі, що моделюється

Для кожної з 200 тестових багатовимірних імпульсних послідовностей було проведено фіксування імпульсної послідовності, яка подавалась на вхід системи, вихід порогового елементу як результат процесу розпізнавання, значення цільового виходу порогового елементу та середнє значення помилки розпізнавання (mae) після навчання нейронної мережі за алгоритмами лінійної

класифікації та зворотного розповсюдження. Середнє значення помилки визначалось як відношення суми модулів різниці цільового та отриманого виходу порогового елементу у кожній точці реєстрації - моментів часу фіксації відповіді порогового елементу - до загальної кількості точок реєстрації. На рис.2 подана відповідна вище вказана інформація щодо 15-ої тестової послідовності, де відповідь порогового елемента, реєстрована через проміжки 0,05с протягом періоду 1с, позначена на графіку синьою лінією, а цільовий вихід – червоною лінією.

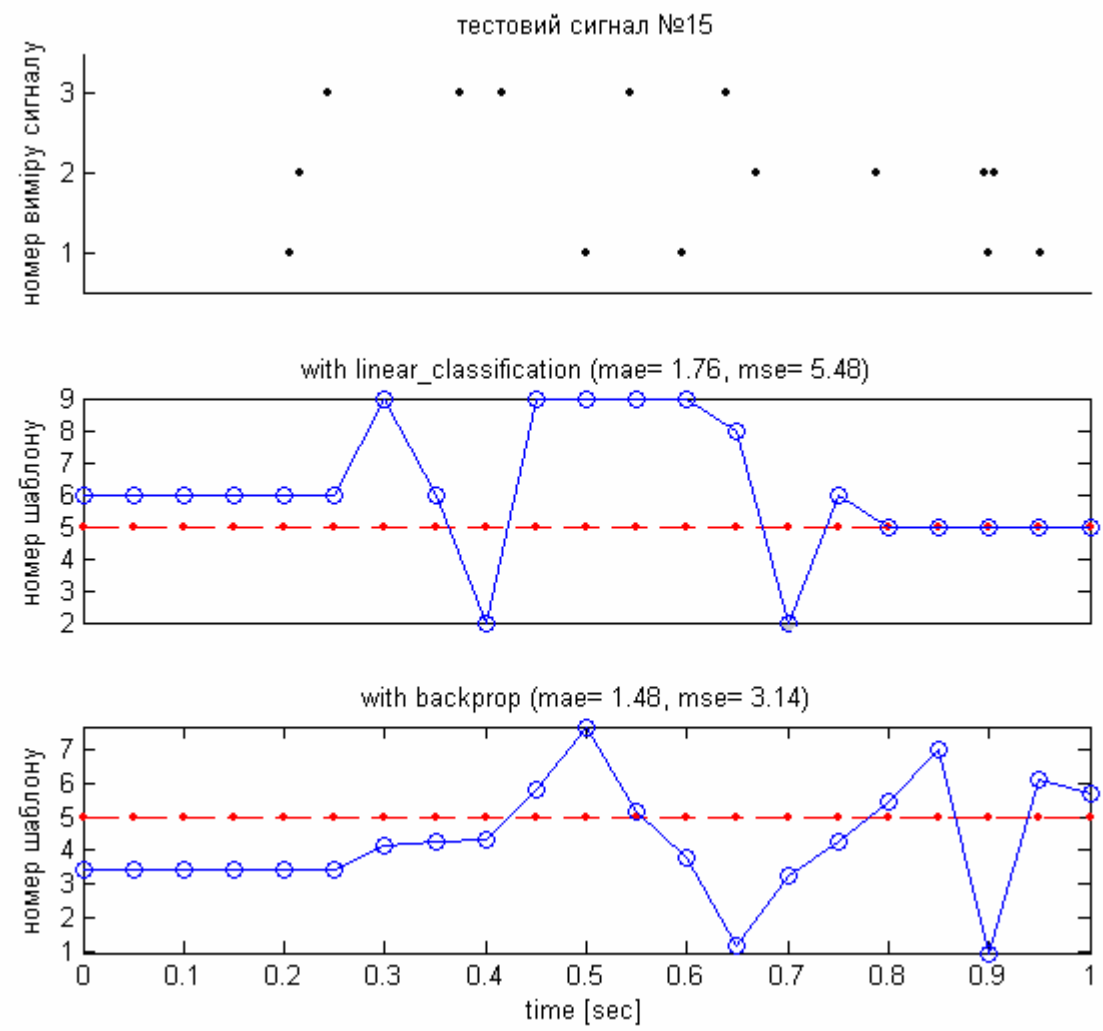


Рисунок 2 - Результат розпізнавання 15-ої тестової 3-вимірної імпульсної послідовності

З 200 тестових імпульсних послідовностей після навчання мережі за алгоритмом лінійної класифікації вірно було розпізнано 161 послідовність (80,5%), а після навчання за алгоритмом зворотного

розповсюдження – 111 (55,5%). Гістограма розподілу відсотку розпізнаних імпульсних послідовностей представлена на рис. 3.

Середнє значення помилки розпізнавання, яка обраховується як відношення кількості точок (моментів часу реєстрації вихідних імпульсів), у яких вихід порогового елемента різниться від цільового, до загальної кількості точок реєстрації, по усім багатовимірним імпульсним послідовностям після навчання мережі за алгоритмом лінійної класифікації складає 3.34, а після навчання за алгоритмом зворотного розповсюдження - 3.63. Гістограма розподілу середнього значення помилки представлена на рис. 3.

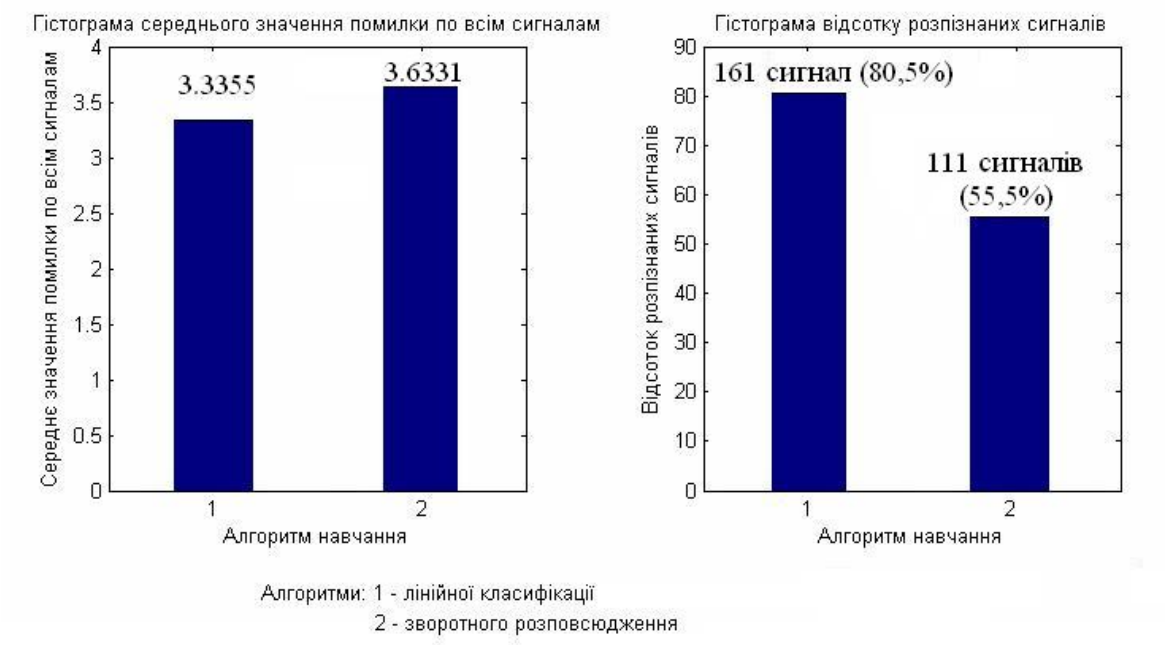


Рисунок 3 - Гістограма середнього значення помилки по всіх тестових імпульсних послідовностях та відсотку розпізнавання тестових імпульсних послідовностей

ВИСНОВКИ

Проведений аналіз результатів дає підстави стверджувати, що кращим алгоритмом навчання спайкінгової нейромережі за критерієм достовірності та значенням помилки розпізнавання є алгоритм лінійної класифікації. Так, після навчання нейронної мережі за вказаним алгоритмом результати розпізнавання багатовимірних імпульсних послідовностей є більш достовірним, а помилка розпізнавання меншою, ніж після навчання мережі за алгоритмом зворотного розповсюдження.

Перспективами в подальшому дослідженні даної проблематики є вибір оптимальної структури мережі, що давала б кращі результати

розпізнавання, дослідження нових алгоритмів навчання, збільшення кількості шаблонів розпізнавання.

ЛІТЕРАТУРА

1. Maass W. Networks of spiking neurons: the third generation of neural network models / W. Maass // *Neural Networks*, 1997, №10. — P. 1659-1671. – ISSN 0893-6080.
2. Бардаченко В. Ф. Таймерні нейронні елементи та структури. Монографія / В. Ф. Бардаченко, О. К. Колесницький, С. А. Василецький. – Вінниця : УНІВЕРСУМ-Вінниця, 2005, 126 с. – ISBN 966-641-109-1.
3. Бардаченко В. Ф. Перспективи застосування імпульсних нейронних мереж з таймерним представленням інформації для розпізнавання динамічних образів / В. Ф. Бардаченко, О. К. Колесницький, С. А. Василецький // УСіМ. – 2003. - №6. - С. 73-82.
4. Kolesnytskyj O. K. Optoelectronic Implementation of Pulsed Neurons and Neural Networks Using Bispin-Devices / O. K. Kolesnytskyj, I. V. Bokotsey, S. S. Yaremchuk // Optical Memory & Neural Networks (Information Optics), 2010, Vol.19, №2, pp.154-165.
5. Колесницький О. К. Принципи побудови архітектури спайкових нейрокомп'ютерів / О. К. Колесницький // Вісник Вінницького політехнічного інституту. – Вінниця: УНІВЕРСУМ-Вінниця. – 2014. – №4 (115), С.70-78. [Електронний ресурс]. Режим доступу - http://visnyk.vntu.edu.ua/article/view/3697/5416
6. Колесницький О. К. Аналітичний огляд апаратних реалізацій спайкових нейронних мереж / О. К. Колесницький // Математичні машини і системи. – 2015. – №1, С.3-19. ISSN 1028-9763 [Електронний ресурс]. Режим доступу - http://www.immsp.kiev.ua/publications/articles/2015/2015_1/01_2015_Kolesnytskyy.pdf
7. CSIM: A Neural Circuit SIMulator Version 1.1 User Manual // The IGI LSM group. - 2006.
8. Wulfram Gerstner and Werner M. Kistler Spiking Neuron Models Single Neurons, Populations, Plasticity // Cambridge University Press. – 2002. – С. 116-119

Отримано 26.06.2017

УДК 004.925.4

РОЗРОБКА МЕТОДІВ ТЕКСТУРУВАННЯ ДЛЯ ЗАДАЧ ФОТОРЕАЛІСТИЧНОГО РЕНДЕРИНГУ

Романюк О. Н., Дудник О. О.
Вінницький національний технічний університет

Запропоновано методи для підвищення продуктивності та реалістичності текстурування, зокрема, методи перспективно-коректного текстурування, фільтрації текстур, рельєфного текстурування.

При формуванні графічних зображень вирішується двоєдина задача – підвищення продуктивності та підвищення реалістичності. Сьогодні продуктивності графічних засобів достатньо для формування зображень, які за своїми візуальними властивостями близькі до фотографій, тобто досягається фотографічна якість.

Базовим підходом до формування фотореалістичних зображень є текстурування, яке полягає у накладанні на поверхню полігонів ділянок існуючих зображень (як правило з фотографій). Для визначення кольору пікселя на екрані встановлюється відповідність між екранним пікселем та точкою на текстурі.

При визначенні текстурних координат важливо врахувати деформацію полігонів при перспективних перетвореннях. Тому при перспективно-коректному текстуруванні, у переважній більшості випадків, текстурні координати визначають за методом Хекберта [1-4]:

$$u_i = \frac{A_1 \cdot x_i + B_1 \cdot y_i + C_1}{D \cdot x_i + E \cdot y_i + F},$$
$$v_i = \frac{A_2 \cdot x_i + B_2 \cdot y_i + C_2}{D \cdot x_i + E \cdot y_i + F}. \qquad (1)$$

де u і v – текстурні координати, x і y – екранні координати об'єкта, A_1, A_2, B_1, B_2, C_1, C_2, D, E, F – коефіцієнти полігона, який текстурується. Як видно з формули (1), знаходження текстурних координат є трудомісткою процедурою, тому було знайдено ітеративні формули для зменшення кількості операцій.

Формула (1) досить складна для обчислення в реальному часі на кожен піксель. Розроблено алгоритм паралельного підрахунку координат текселів: для кожного рядка растеризації спочатку обчислюються параметри u_t і u_b, після чого паралельно обчислюються значення чисельника за формулою [2]:

$$w_n = u_t + A_1 \cdot n,$$

і знаменника за формулою

$$v_n = u_b + D \cdot n,$$

після чого виконується операція ділення для визначення координати:

$$u_n = \frac{w_n}{v_n}.$$

Оскільки для кожного наступного x, n збільшується на 1, має місце така формула [1]:

$$w_n = w_{n-1} + A_1, \quad v_n = v_{n-1} + D. \tag{2}$$

Таким чином, відповідно до формул (2) можна запропонувати послідовну процедуру обчислення текстурних координат: w_n і v_n для кожного x обчислюються шляхом додавання до w_{n-1} і v_{n-1}, визначених для x-1, A_1 і D відповідно [2].

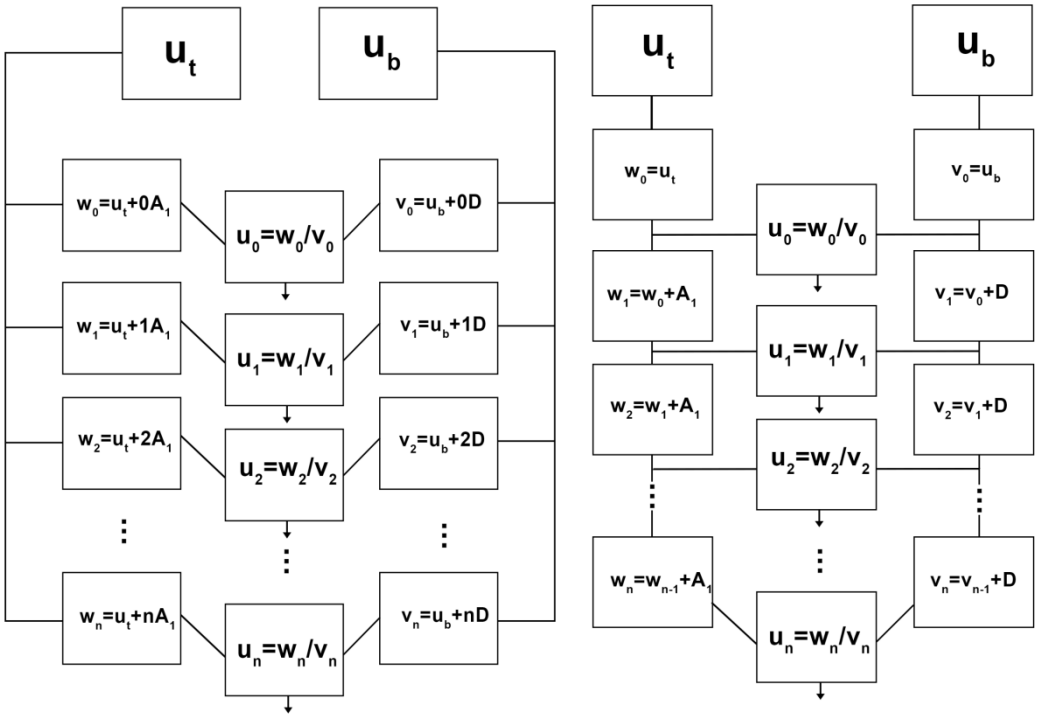


Рисунок 1 – Паралельний (зліва) та послідовний (зправа) підрахунок координат текселів

Різниця в рівнях деталізації ділянки зображення на екрані та на текстурі призводить до появи аліайзингу, тому при накладанні текстур виконують фільтрацію. Серед поширених методів фільтрації є білінійна та трилінійна.

При білінійній фільтрації значення кольору пікселя обраховується шляхом білінійної інтерполяції значень кольорів чотирьох текселів. При трилінійній фільтрації виконують лінійну інтерполяцію результатів білінійної фільтрації на двох сусідніх MIP-рівнях. На жаль, ці методи не забезпечують високу реалістичність та

призводять до втрати деталей зображення, оскільки вони не враховують положення полігонів відносно спостерігача.

Найвищого рівня реалістичності дозволяє досягти анізотропна фільтрація. При анізотропної фільтрації проекція пікселя на поверхню текстури розглядається, як витягнутий еліпс, що дозволяє точніше визначати кольори пікселів (рис 2).

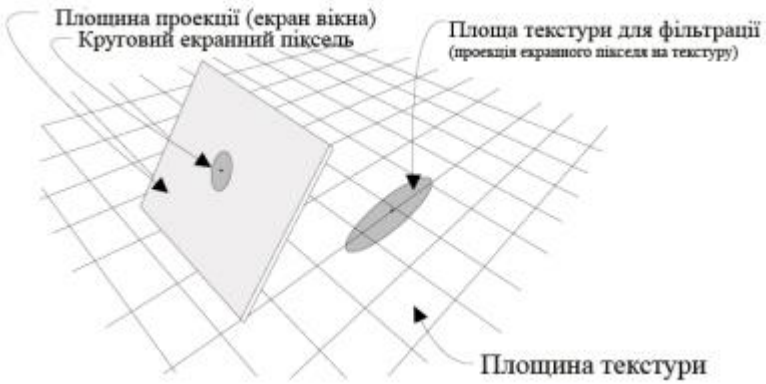


Рисунок 2 – Проекція пікселя на текстуру

Для визначення необхідних для фільтрації текселів складають рівняння еліпса за алгоритмом [3-4]:

1. Визначення векторів осей:

$$(U_x, V_x) = \left(\frac{du}{dx}, \frac{dv}{dx}\right),$$
$$(U_y, V_y) = \left(\frac{du}{dy}, \frac{dv}{dy}\right),$$

2. Визначеня коефіцієнтів рівняння:

$$A = V_x^2 + V_y^2, \quad B = -2(U_x V_x + U_y V_y),$$
$$C = U_x^2 + U_y^2, \quad F = (U_x V_y + U_y V_x)^2,$$

3. Кінцеве рівняння:

$$AU^2 + BUV + CV^2 = F,$$

де $U = u - u_0, V = v - v_0$.

Похідні знаходять за формулами:

$$\frac{\partial u}{\partial x} = u(x+1, y) - u(x, y) = \frac{A(x+1) + By + C}{G(x+1) + Hy + I} - \frac{Ax + By + C}{Gx + Hy + I},$$
$$\frac{\partial v}{\partial x} = v(x+1, y) - v(x, y) = \frac{D(x+1) + Ey + F}{G(x+1) + Hy + I} - \frac{Dx + Ey + F}{Gx + Hy + I},$$
$$\frac{\partial u}{\partial y} = u(x, y+1) - u(x, y) = \frac{Ax + B(y+1) + C}{Gx + H(y+1) + I} - \frac{Ax + By + C}{Gx + Hy + I},$$
$$\frac{\partial v}{\partial y} = v(x, y+1) - v(x, y) = \frac{Dx + E(y+1) + F}{Gx + H(y+1) + I} - \frac{Dx + Ey + F}{Gx + Hy + I}.$$

Як видно з наведених формул обчислення часткових похідних для визначення коефіцієнтів рівняння еліпса є трудомісткою процедурою. На кожен піксель в рядку растеризації необхідно виконати 10 операцій множення, 10 операцій додавання, 4 операції ділення та 1 операцію віднімання. Тому запропоновано ітераційні формули для спрощення процедури визначення похідних [2]:

$$\frac{\partial u}{\partial x} = \frac{num_{i-1} + M}{(K + Gx)(L + Gx)}, \frac{\partial v}{\partial x} = \frac{num_{i-1} + N}{(K + Gx)(L + Gx)},$$
$$\frac{\partial u}{\partial y} = \frac{num_{i-1} + O}{(K + Gx)(L + Gx)}, \frac{\partial v}{\partial y} = \frac{num_{i-1} + P}{(K + Gx)(L + Gx)}.$$

де $J = G + I$, $K = I + Hy$, $L = G + I + Hy = J + Hy$, а параметри N, M, O, P визначаються за формулами:

$$M = AH - BG, \quad N = DH - EG, O = BG - AH, P = EG - DH .$$

Таким чином, кількість операцій додавання зменшено на 4, операцій множення зменшено на 8, а операцій віднімання зменшено на 1 [2].

Встановлено, що частина проміжних результатів обчислень при визначенні параметрів рівняння еліпса може бути закешована в оперативній пам'яті для повторного використання без перерахунку, що дозволяє підвищити продуктивність обчислень до 70% [3].

Для визначення кольору екранного пікселя використовують усереднення кольору всіх текселів, які попадають в еліпс. При усередненні кольорів текселів можуть використовуватись вагові коефіцієнти, що дозволяють врахувати фізичні особливості пікселів пристроїв виводу. В переважній більшості існуючих засобів комп'ютерної графіки використовують вагові коефіцієнти, які не залежать від розмірів та форми проекції пікселя на текстуру [5-8]. Тому запропоновано вагову функцію на основі Гаусівської моделі пікселя (рис 4), яка дає можливість точніше визначати кольори пікселів [5]:

$$W(u,v) = \left(1 - \frac{-1}{2}\left[\frac{\left[U \cdot \cos(\theta) + V \cdot \sin(\theta)\right]^2}{\sigma_1^2} + \frac{\left[V \cdot \cos(\theta) - U \cdot \sin(\theta)\right]^2}{\sigma_2^2}\right]\right)^8$$

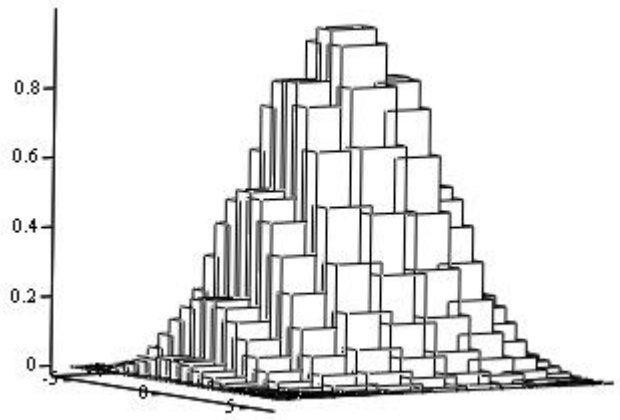


Рисунок 3 – Графік вагової фонкції

Висока обчислювальна складність ускладнює використання вагових функцій на основі гаусівської моделі пікселя у складних графічних сценах із високими вимогами до продуктивності обчислень. Тому запропоновано метод анізотропної фільтрації з використанням спеціальних текстурних карт вагових коефіцієнтів. Застосування карт вагових коефіцієнтів дає можливість істотно підвищити продуктивність анізотропної фільтрації із використанням складних вагових функцій за рахунок використання заздалегідь розрахованих вагових коефіцієнтів.

Для реалістичного відтворення нерівностей використовують різні методи рельєфного текстурування. Найбільш ефективним вважають parallax mapping. Parallax mapping використовується для процедурного створення тривимірного опису текстурованої поверхні з використанням карт висот замість генерації нової геометрії. Parallax mapping виконують шляхом зміщенням текстурних координат таким чином, щоб поверхня здавалася об'мною. Основний принцип методу полягає в тому, щоб визначати колір пікселя за текстурними координати тієї точці, де видовий вектор перетинає поверхню. Зміщення текстурних координат визначають шляхом прорахунку трасування променів для карти висот.

Виконання анізотропної фільтрації при текстуруванні з використанням техніки parallax mapping може призводити до некоректного визначення кольорів пікселів, надмірного розмиття окремих ділянок зображення. Задача полягає в тому, що відомі методи анізотропної фільтрації не враховують нерівності поверхні, а тому всі текселі, що формують проекцію піксела на текстуру в формі еліпса, вважаються видимими та використовуються для визначення кольору піксела. Насправді ділянки з більшою висотою можуть перекривати ділянки з меншою висотою.

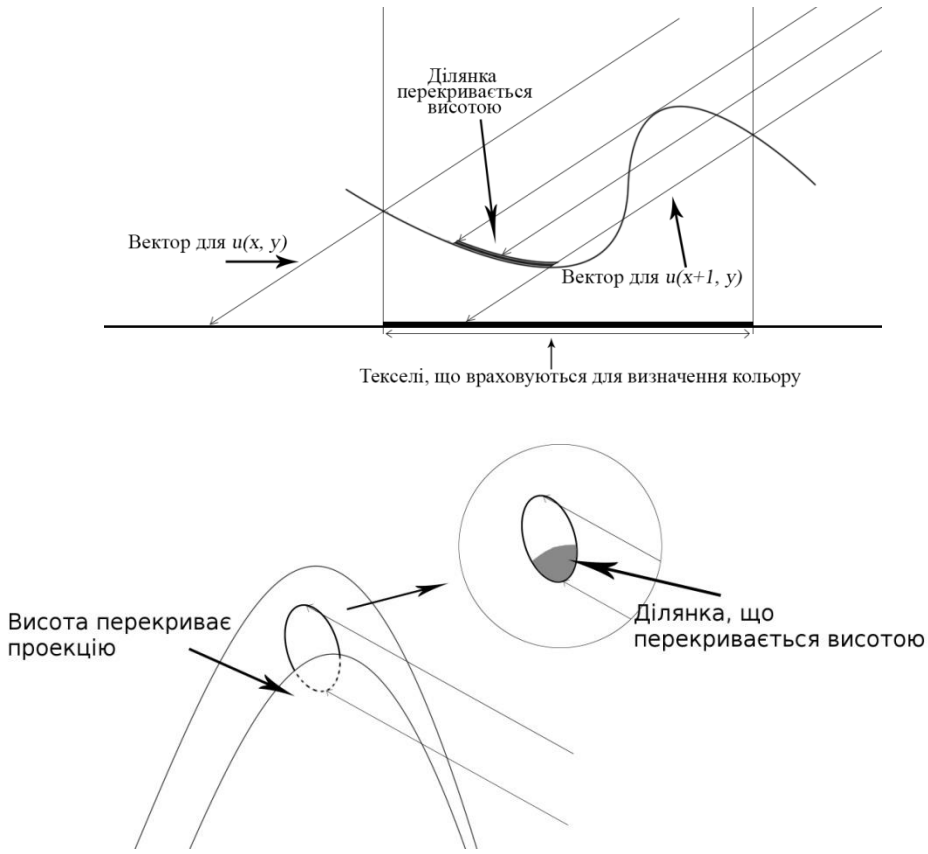


Рисунок 4 – Перекриття частини проекції пікселя більшою висотою

Для усунення артефактів запропоновано метод врахування рельєфних особливостей поверхні при анізотропній фільтрації текстур шляхом зворотного трасування променя для визначення видимості точки.

Трасування видового вектора – достатньо ресурсомістка процедура, а тому потребує використання спрощених методів обчислень для забезпечення високої продуктивності при відтворенні графічних зображень у реальному часі. Високу точність і достатню продуктивність забезпечує метод Доннеллі, що базується на використанні додаткової тривимірної текстури – карти відстаней до поверхні. Ця текстура будується на основі карти висот, де в якості третього виміру використовується висота. При цьому значення кожного текселя є найближча відстань до поверхні, заданої картою висот [6].

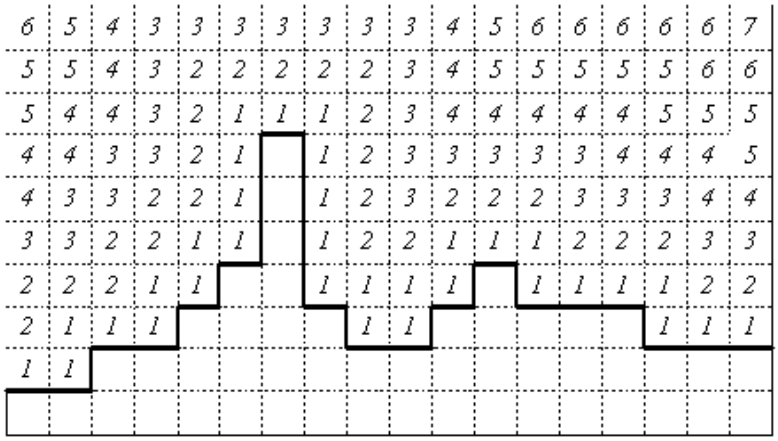


Рисунок 5 – Двовимірний фрагмент карти відстаней до поверхні

Було доведено, що метод Донеллі не є оптимальним для поверхонь із складним рельєфом. Тому запропоновано модифікацію метода Донеллі, яка полягає в уточненні карти відстаней до поверхні для виконання parallax mapping. Вектора, який проходить через деяку точку P, що лежить над поверхнею, і перетинає поверхню таким чином, що частина вектора на проміжку від точки P до поверхні рівна мінімальній відстані від цієї точки до поверхні, і, при цьому, не перетинає поверхню на проміжку від початку вектора до цієї точки P може не існувати. В такому випадку зберігати в карті відстаней мінімальну відстань від точки P до поверхні не доцільно. Оптимальніше зберігати найменшу з можливих відстаней від точки до поверхні за умови, що вектор, який проходить через точку P, не перетинається з поверхнею на проміжку від початку вектора до точки P. Уточнення дає можливість знизити обчислювальні затрати при текстуруванні поверхонь із складним рельєфом без втрати точності обчислень.

ВИСНОВКИ

Запропоновано методи для підвищення продуктивності та реалістичності текстурування, зокрема, методи перспективно-коректного текстурування, фільтрації текстур, рельєфного текстурування.

ЛІТЕРАТУРА

1. Романюк О. Н. Неортогональна растеризація при перспективнокоректному текстуруванні [Текст] / О. Н. Романюк, О. О. Дудник, О. В. Мельник // VI Международная конференция «Моделирование и компьютерная графика», м. Красноармійськ, 25-29 травня 2015 р. — 2015. — С. 174-178.

2. Романюк О. Н. Метод підвищення продуктивності перспективно-коректного текстурування [Текст] / О. Н. Романюк, О. О. Дудник // Наукові праці

ДонНТУ. Серія "Інформатика, кібернетика та обчислювальна техніка". - 2016. - № 1 (22). - С. 43-46.

3. Романюк О. Н. Підвищення продуктивності перспективно-коректного текстурування з використанням анізотропної фільтрації [Текст] / О. Н. Романюк, О. О. Дудник // Вимірювальна та обчислювальна техніка в технологічних процесах. - 2016. - № 3. - С. 192-195.

4. Романюк О. Н. Анізотропна фільтрація текстур з використанням методів кешування [Текст] / О. Н. Романюк, О. О. Дудник // Вісник Вінницького політехнічного інституту. - 2016. - № 6. - С. 59-64.

5. Романюк О. Н. Анізотропна фільтрація з використанням вагової функції на основі Гаусівської моделі пікселя [Текст] / О. Н. Романюк, І. В. Абрамчук, О. О. Дудник // Вимірювальна та обчислювальна техніка в технологічних процесах. - 2016. - № 2. - С. 117-121.

6. Романюк С. О. Анізотропна фільтрація з використанням вагових функцій / С. О. Романюк, О. О. Дудник, Л. А. Савицька, О. В. Романюк // Вісник Херсонського національного технічного університету. – Херсон – 2015. - № 3. - С. 459-46

7. Романюк О. Н. Математичні моделі піксела / О. Н. Романюк, О. О. Дудник // Електронні інформаційні ресурси: створення, використання доступ. Збірник міжнародної ноуково-практичної конференції. - Вінниця. - 2014. - С. 289-293.

8. Романюк О. Н. Модифікація гаусівської моделі піксела для задач антиаліайзингу [Текст] / О. Н. Романюк, І. В. Абрамчук, О. О. Дудник, О. В. Мельник // Наукові праці ДонНТУ. Серія "Інформатика, кібернетика та обчислювальна техніка". — 2015. - № 1(20). — С. 84-88.

9. Романюк О. Н. Особливості анізотропної фільтрації текстур при використанні технології parallax mapping [Текст] / О. Н. Романюк, О. О. Дудник // Вісник хмельницького національного університету серія: Технічні науки. - 2017. - № 1. - С. 236-241.

10. Романюк О. Н. Модифікований метод parallax mapping з використанням карти відстаней до поверхні [Текст] / О. Н. Романюк, О. О. Дудник, О. В. Романюк // Інформаційні технології та комп'ютерна інженерія. - 2017. - № 1. - С. 78-82.

Отримано 26.06.2017

Секція 1. Моделювання систем

УДК 519.226

ПОБУДОВА СЦЕНАРІЇВ ІЗ ВИКОРИСТАННЯМ БАЙЄСІВСЬКИХ МЕТОДІВ

В.Г. Гуськова, П.І. Бідюк
НТУУ «Київський політехнічний інститут імені Ігоря Сікорського», Україна

Наведено існуючи сучасні підходи та методики побудови сценаріїв для моделювання процесів різноманітної природи. Розглянуто основні типи сценаріїв та можливості їх застосування. Проаналізовано вимоги, яких треба дотримуватися при розробці сценаріїв для розв'язання задач та визначення можливості застосування сценарного підходу разом і мережами Байєса.

Ключові слова: сценарії, сценарній підхід, сценарне планування, методики, мережі Байєса.

1. ВСТУП

Сценарне планування є одним із способів довгострокового планування, яке використовується у більшості західних компаній протягом останніх років. Розробка сценаріїв використовується як альтернативний підхід до лінійного планування, застосування якого дуже часто призводить до неточності та неефективності при побудові різноманітних прогнозів та планів. Головним інструментом сценарного планування є сценарний аналіз, який застосовується для управління стратегічними процесами з високим або середнім рівнем невизначеності. Такий метод може застосовуватися, якщо ймовірність виникнення подій однакова та необхідно завчасно обрати стратегію дії в межах кожного із сценаріїв.

2. ВИДИ СЦЕНАРІЇВ ТА УМОВИ ЇХ ЗАСТОСУВАННЯ

Сценарії можуть застосовуватися з метою досягнення різних цілей. В залежності від поставленої мети існують типи сценаріїв:

1. Візуальний опис бажаного. Використовується в разі потреби у розробці характеристики найбільш бажаного результату.
2. Проектний – формулювання припущень щодо очікуваного майбутнього. Використовується у аналізі можливих наслідків поточної або запланованої діяльності.

3. Маршрутний – визначення шляху досягнення майбутнього стану з теперішнім із використанням порівняння теперішнього із бажаним сценарієм. Використовується для розуміння послідовності процесу.

4. Альтернативний – порівняння варіантів із використанням багатьох сценаріїв візуального, проектного та маршрутного типів. Використовується для створення знань для багатьох можливих станів.

Для зв'язку значень перехресного впливу з елементами матриці використовують експертний висновок, значення оцінок яких встановлюється в діапазоні від 0 до 1. Байєсівські ймовірності можуть бути використані при сценарному підході як генеруючий інструмент, а також як інструмент для визначення того, які із сценаріїв є більш імовірними на основі знань, наявних на даний момент. Вони оцінюються на основі явних подій, що можуть бути певним чином кластеризовані та поділені на повні набори умов, що називаються «сценаріями». У свою чергу, байєсівські ймовірності постійно навчаються по мірі надходження нової інформації аж до моменту досягнення фіналу у майбутньому. Таким чином, байєсівські ймовірності не суперечать описовій, якісній (а не кількісній) природі сценарного підходу, оскільки не є прямою проекцією минулого на майбутнє. Використання байєсівських ймовірностей спонукає учасників процесу побудови сценаріїв обґрунтовувати власні судження, виявляючи, таким чином, приховані припущення, альтернативи та передбачення, які часто не озвучуються і випускаються із розгляду.

3. ВИКОРИСТАННЯ ЙМОВІРНОСТЕЙ У СЦЕНАРІЯХ

Множина сценаріїв, згідно апостеріорних ймовірностей їх виникнення, складається з можливих сценаріїв та найбільш бажаних сценаріїв, тобто візуальних. Аналітичні сценарії, побудовані на основі моделювання, аналізу перехресних впливів та байєсівських ймовірностей, дозволяють краще зрозуміти умовну природу процесів та визначити дії, які потрібно здійснити для того, щоб зробити імовірним найбільш бажаний сценарій. Таким чином, застосування апарату ймовірностей робить стратегічне планування більш успішним завдяки дослідженню різноманітних умов перебігу процесу.

Використання аналізу перехресних впливів дають можливість моделювання можливих небажаних подій у майбутньому. Таке моделювання неможливо здійснити із використанням самого лише інтуїтивного підходу до побудови сценаріїв.

Введемо у модель експертні знання про те, що деякі змінні є залежними або незалежними між собою, відтак отримаємо первинні зв'язки мережі Байєса (БМ). Таким чином, вибір варіанта та значення залежить від X_1, X_2, X_3 та X_4. Для прийняття рішення по певному X_i ми обчислюємо ймовірності значень x_i, що входять до X_i. Це дає значення ймовірності:

$$p(x_i|C) = \sum_{x_1} \dots \sum_{x_{i-1}} \sum_{x_{i+1}} \dots \sum_{x_n} p(x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n|C)$$

Необхідно здійснити ініціалізацію випадкових змінних. Надамо випадковій змінній X_k $(i < k < n)$ значення $x_{ko} \in \{x_{k1}, \dots, x_{kmk}\}$. Застосовуючи нормуючу константу α, отримуємо

$$p(x_i|C) = \alpha \sum_{x_1} p(x_1|C) \sum_{x_2} p(x_2|x_1, C) \dots \sum_{x_{i-1}} p(x_{i-1}|x_1, \dots, x_{i-2}, C)\ p(x_i|x_1, \dots, x_{i-1}, C) \cdot$$
$$\cdot \sum_{x_{i+1}} p(x_{i+1}|x_1, \dots, x_i, C) \dots \sum_{x_{k-1}} p(x_{k-1}|x_1, \dots, x_{k-2}, C)\ p\left(x_{k_0}|x_1, \dots, x_{k-1}, C\right)$$

Здійснимо заміну із використанням функції подібності $\lambda(x_1, \dots, x_i)$

$$\lambda(x_1, \dots, x_{k-1}) = p(x_{k_0}|x_1, \dots, x_{k-1}, C).$$

Таким чином, надаючи змінним різні значення, отримуємо модель сценарію для розв'язання поставлених задач за допомогою байєсівського підходу.

4. ВИСНОВКИ

Підсумовуючи матеріал можна сказати, що доцільно використовувати байєсівські ймовірності у сценарному моделювання. Також треба зазначити, що успішність такого підходу має ряд чинників, які повинні бути дотримані. Якщо застосування ймовірностей не суперечить вирішенню поставленої задачі, то їх треба використовувати. Але даний підхід потребує більше часу та має вищу загальну складність.

СПИСОК ЛІТЕРАТУРИ:

[1] **Malakoff D.** *Bayes Offers a `New' Way to Make Sense of Numbers.(Bayesian statistics)* [Електронний ресурс] / David Malakoff // Режим доступу: http://socsci2.ucsd.edu/~aronatas/project/academic /Bayesian%20statistics%20Malakoff%20in%20Science.pdf

[2] **Бидюк П.И., Терентьев А.Н., Гасанов А.С**. *Построение и методы обучения Байесовских сетей* [Текст] / Кибернетика и системный анализ. - 2005. – №4. – с. 133-147.

Отримано 20.06.2017

УДК 004.032.24

ОСОБЕННОСТИ ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ ДЛЯ ВЫПОЛНЕНИЯ В РАСПРЕДЕЛЕННОЙ СРЕДЕ

О.А.Дмитриева, О.Л.Половинка
Донецкий национальный технический университет,
Украина, Покровск

У статті дано визначення поняттю даних великого обсягу, наведено огляд існуючих засобів для їх обробки, доведена необхідність дослідження та розробки паралельних алгоритмів для аналізу великих обсягів даних, сформульовано етапи розпаралелювання алгоритмів інтелектуального аналізу.

Ключові слова: Big Data, великі обсяги даних, Data Mining, інтелектуальний аналіз даних, обробка даних, паралельні алгоритми.

В реалиях современного мира, эффективность функционирования любого предприятия, в первую очередь, зависит от обработки и анализа собираемых и накапливаемых в процессе его работы данных. Со временем эти данные занимают огромные объемы памяти, и для их хранения используются различные технические средства. При этом источники, в которых они находятся, могут быть разрозненными, а сами данные неструктурированными или частично структурированными.

Накопленная информация важна для многих предприятий, однако сложность ее анализа постоянно увеличивается, а также возрастает стоимость обработки данных. Методы, алгоритмические модели и средства их реализации, используемые на этапах анализа, сбора и обработки больших объемов данных постоянно совершенствуются и развиваются одновременно с развитием компьютерных технологий и теоретических основ анализа данных. Поэтому выбранная тематика современна и актуальна.

Цель работы - формулировка понятия «большие данные», анализ методов и средств распараллеливания алгоритмов интеллектуального анализа данных, выбор дальнейших этапов исследования.

Совокупность методов и технологий обработки структурированных и неструктурированных данных больших объемов получила название Big Data. Интерес к термину «большие данные» или Big Data проявился в 2012 году и постоянно возрастает. Это наглядно видно на рисунке 1.

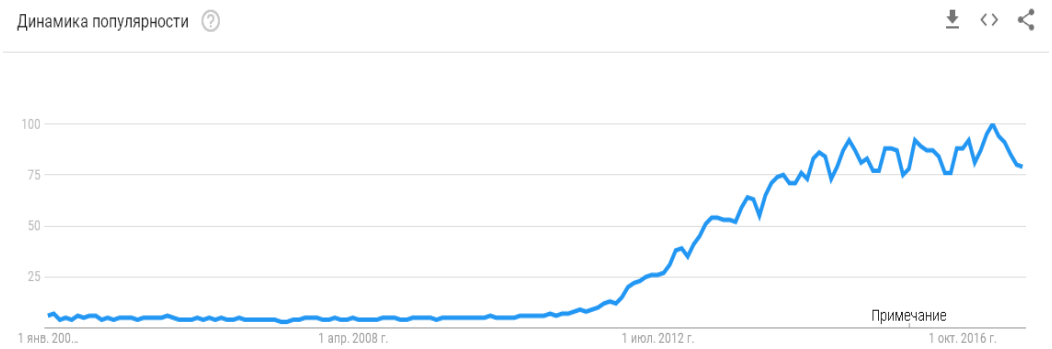


Рисунок 1 - Динамика запросов по «Big Data» от Google с 2004 по 2017 год

Итак, Big Data- это совокупность технологий, предназначенных для выполнения следующих операций:

– обработки больших объемов данных, занимающих терабайты памяти;

– анализа постоянно поступающих новых данных;

– работы с неструктурированными данными, извлечения из них различной информации и структуризации ее в процессе построения различных аналитик [1].

Термин Big Data связан с понятием базы данных (БД), но значительно превосходит указанную структуру по возможностям занесения, хранения, управления и анализа информации. В таблице 1 приведены основные отличия обычных баз данных от баз данных больших объемов [2].

Таблица 1. Сравнение Традиционной базы данных и базы Big Data

Характеристики	**Стандартная БД**	**БД большого объема**
Объем данных	1Гбайт – сотни терабайт	От петабайт до эксабайт (1 Эксабайт = 2^{60} байт)
Способ хранения	Централизованный	Децентрализованный
Структура данных	Структурированы	Полу-структурированные или неструктурированные
Хранение и обработка (модель)	Вертикальная модель	Горизонтальная модель
Взаимосвязь	Сильная	Слабая

Задачи сбора, хранения и обработки являются традиционными при работе с большими объемами данных. Кроме того, рассматривают также и технологии создания хранилищ данных.

Для обработки информации используются методы оперативного (OLAP) и интеллектуального анализа (ИАД, Data Mining). А для определенных типов данных выделяют методы визуального и текстового анализа (Visual Mining, Text Mining).

Однако, один лишь факт наличия данных не гарантирует получение результатов, повышающих рабочие показатели. Нужно суметь преобразовать "сырые" данные в полезную для принятия важных бизнес решений информацию. В этом и заключается одна из основных задач технологии Data Mining [3].

Перечислим основные причины популярности данной технологии:

– возможность анализировать большие массивы прежде накопленной и вновь собираемой информации;

– существенное увеличение объективности получаемых данных, которое достигается исключением или снижением роли человеческого фактора на этапе обработки и анализа информации;

– дешевизна (внедрение технологий Data Mining является более дешевым, в сравнении с необходимостью содержать постоянный штат высококвалифицированных специалистов).

Современной тенденцией увеличения эффективности алгоритмов Data Mining является построение параллельных алгоритмов. На рисунке 2 приведена систематизация подходов к распараллеливанию алгоритмов интеллектуального анализа данных.



Рисунок 2- Систематизация параллельных алгоритмов интеллектуального анализа данных

Можно выделить два основных подхода к разработке параллельных алгоритмов ИАД:

– декомпозиция алгоритма с целью распараллеливания отдельных блоков;

– индивидуальное распараллеливание.

Декомпозиция - общий подход, заключающийся в разделении алгоритма ИАД на части, которые могут выполняться параллельно.

При индивидуальном распараллеливании алгоритмов используется индивидуальный подход к каждому алгоритму ИАД, а в дальнейшем выбирается эффективная в заданных условиях параллельная структура [4].

Наиболее активные разработки ведутся в рамках индивидуального подхода к распараллеливанию ИАД. Они, в свою очередь, имеют два направления: параллельный и распределенный интеллектуальный анализ данных. Однако, большинство разработок направлено на оптимизацию распараллеливания отдельных алгоритмов в конкретной заданной среде, что зачастую приводит к невозможности их использования в других условиях. Следовательно, этот подход не может рассматриваться как общий для построения параллельных алгоритмов ИАД.

В связи с вышесказанным, для разработки параллельных алгоритмов предлагается использование иерархической декомпозиционной методики, аналогичной описанной в [5]. Методика состоит в том, что основная задача разбивается на подзадачи. Анализируются информационные связи между задачами, определяются необходимые операции, оценивается эффективность параллелизма. Если на каком-либо этапе получены неудовлетворительные результаты, процесс останавливается, изменяется схема разбивки на подзадачи и все этапы повторяются с новыми начальными условиями. Распараллеливание алгоритма и его оценка повторяется до тех пор, пока не будут получены удовлетворительные характеристики параллельного алгоритма, или пока невозможность добиться качественного распараллеливания не станет очевидной. Основной оценочной характеристикой качества выполняемого алгоритма, при этом, являются временные параметры.

Для минимизации воздействия среды на эффективность алгоритма, необходимо разработать структуру линий связи, которая должна удовлетворять следующим требованиям:

– обеспечивать связь между двумя любыми процессорами или модулями;

– поддерживать максимальное количество одновременных связей.

Предлагается ориентация разрабатываемых алгоритмов на использование в многопроцессорных вычислительных системах SIMD, MIMD и кластерной архитектуры с различными топологиями соединения процессорных элементов.

По итогам проведенного исследования можно сделать вывод, что единого подхода к созданию параллельных алгоритмов ИАД не существует. Выбранная тематика открывает пласт научно-исследовательской работы, включающей анализ существующих подходов к созданию ИАД; разработку формальной модели алгоритма; выбор методов и построение методики описания параллельных алгоритмов ИАД, выполняемых в распределенной среде; разработку программных модулей для реализации параллельных алгоритмов ИАД; получение оценочных характеристик качества алгоритмов, построенных в соответствии с предложенной методикой. Перечисленное является описанием этапов исследования, результаты которого предполагается апробировать на реальных данных.

ЛИТЕРАТУРА:

[1] DIS Group. Big Data (биг дата) – обработка больших объемов данных [электронный ресурс] http://www.dis-group.ru/solutions/data_management/big_data/

[2] TAdviser – портал выбора технологий и поставщиков [электронный ресурс] http://www.tadviser.ru/

[3] Введение в современный Data Mining [электронный ресурс] http://statistica.ru/local-portals/data-mining/

[4] **Каршиев З.А.**Средства создания параллельных алгоритмов интеллектуального анализа данных /Автореферат диссертации на соискание ученой степени кандидата тех. наук. – СПб, 2013. –18с.

[5] **Л. П. Фельдман, И. А. Назарова** Параллельные алгоритмы численного решения задачи Коши для систем обыкновенных дифференциальных уравнений /Матем. моделирование, 18:9 (2006), 17–31с.

Отримано 27.06.2017

УДК 004.942

ПРОЕКТУВАННЯ ПРОГРАМНОЇ СИСТЕМИ МОДЕЛЮВАННЯ ВЗАЄМОДІЇ ПРОСТОРОВИХ СТРУКТУР

В.А. Святний, Н.Г. Гуськова
Донецький національний технічний університет, м. Покровськ

Розроблено програмну систему моделювання взаємодії просторових структур, в якій виділено блоки, що забезпечують формування початкових даних, задають початкові і граничні умови, обирають схему дискретизації з відповідними додатковими умовами, що пов'язані з точністю апроксимації, виконанням обмежень на крок інтегрування. Для обробки результатів передбачено блок візуалізації.

Ключові слова: математична модель, актин, міозин, біохімічний цикл, концентрація, напруга, взаємодія, чутливість

1. ВСТУП

Дана робота є подовженням досліджень, що наведені в [1-2], і присвячена проектуванню та реалізації програмної системи, яка спрямована на моделювання взаємодії актин-міозин просторових структур у ізотропному в'язкопружному середовищі [3-4] і дослідження залежності між концентрацією АТФ в клітинах і величиною сили, що виникає під час реакції [5]. При реалізації програмної системи передбачено введення додаткової стадії, яка описує процеси розриву ниток актину під вагою головок міозину, що дозволило значно посилити адекватність моделі [2]. Введення агрегованих величин концентрацій зламаних і незламаних ниток актину значно скоротило розмірність системи рівнянь, не втративши при цьому точність чисельних розв'язань, що характеризують реакції взаємодії.

2. ОБҐРУНТУВАННЯ ВИБОРУ СХЕМ ДИСКРЕТИЗАЦІЇ РІВНЯНЬ

При побудові програмної системи серед досліджуваних схем було розглянуто переваги та недоліки декількох схем дискретизації явного і неявного характеру, деякі з них увійшли до програмного комплексу, а саме: в програмній системі було використано центральна схема дискретизації CDS (central difference scheme), перевагою якої є

простота реалізації, швидкість збіжності та можливість обчислення значення функції у точках, що знаходяться на досліджуваному проміжку значень. На рис. 1 наведено схему обчислення значення в точці з використанням центральної схеми дискретизації.

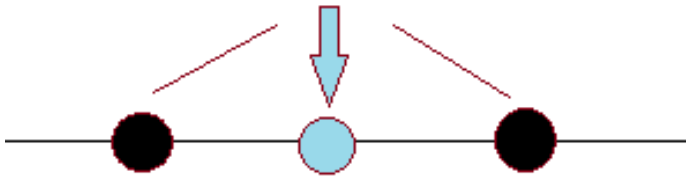


Рисунок 1 - Обчислення значення в точці з використанням центральної схеми

Недоліком центральної схеми дискретизації є неможливість отримання за її допомогою значень в точках, що лежать не всередині проміжку, а на його границях. Цей недолік не дозволяє використовувати схему для пошуку розв'язань на всьому інтервалі. Для уникнення цих проблем було застосовано BDS (Backward difference scheme) (рис. 2) та FDSd (Forward difference scheme) (рис. 3) схеми для дискретизації рівнянь, особливість яких полягає у використанні граничних умов для вирішення рівнянь [6]. Таким чином, для кожного рівняння було складено початкові та граничні умови, які не лише дозволяють використовувати вказані схеми, але й надають рівнянням фізичний сенс.

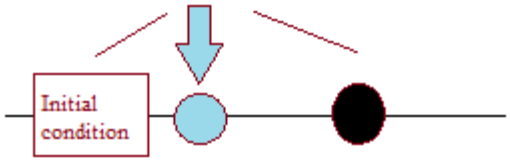


Рисунок 2 - Використання Forward схеми для обчислень

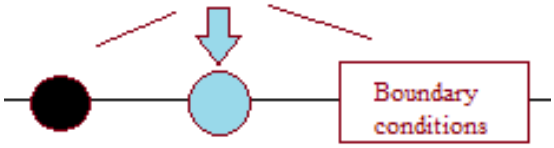


Рисунок 3 - Використання Backward схеми для обчислень

Під час дослідження також було проведено аналіз можливостей використання UpWind схеми дискретизації, яка також має назву «Схема з різницями проти потоку». В ході виконання роботи було визначено, що найбільший ефект дає сумісне використання усіх описаних схем для дискретизації рівнянь. Таким чином, для вирішення поставленої задачі використовувалися усі зазначені схеми. Значення функції у точках, що знаходяться всередині проміжку було

обчислено за допомогою центральної схеми дискретизації, а значення точок, що лежать на початку та кінці проміжків – з використанням початкових та граничних умов рівняння.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ ВЗАЄМОДІЇ

Програмна система складається з декількох функціональних блоків, кожен з яких реалізує один з етапів моделювання. Фіксований інтерфейс обміну даними між ними дозволяє зробити незалежну модифікацію кожного блоку. В програмній системі виділено блоки, що забезпечують формування початкових даних, задають початкові і граничні умови відповідно до обраного типу системи, що моделюється.

Рівняння, що використовуються для визначення концентрації кожної речовини під час спостереження реакції, взаємопов'язані. У звя'язку з використанням різнецевих схем обчислення, розв'язок рівнянь для кожної точки залежить від результатів попередніх експериментів. Загалом, під час вирішення встановленої задачі для знаходження концентрації в окремій точці необхідно мати значення функції на попередньому кроці та граничні умови. Для розрахунку концентрації кожної із змінних використовуються значення інших членів рівняння. Детальну схему розрахунків наведено на рис. 4.

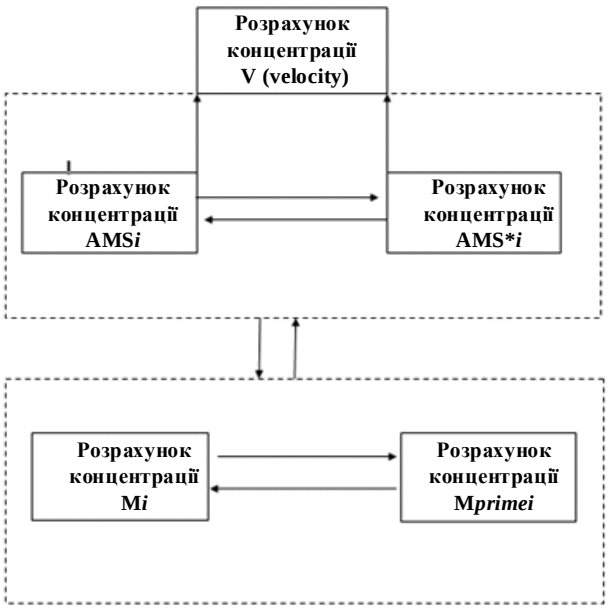


Рисунок 4 - Схема визначення концентрацій речовини на стадіях реакції

На підставі обраної схеми та додаткових умов, що пов'язані з точністю апроксимації, виконанням обмежень на крок інтегрування [7], початковими і граничними умовами, заданими параметрами моделі, наступний блок здійснює розрахунок. Отримані результати передаються у блок для візуалізації результатів у вигляді графіків та таблиць параметрів. У тому разі, якщо будь який з показників, що досліджуються, вийшов за межі припустимих оцінок, проводиться зміна відповідного параметру (параметрів) та процес моделювання повторюється. На підставі отриманих результатів визначаються оптимальні концентрації речовин, показники параметрів, порівнюються час моделювання і розбіжності у результатах.

У початковій формі (рис. 4) програмного інтерфейсу для формування розрахункової схеми необхідно задати кількість розрахункових точок, обрати закон розподілу, визначити початкові значення параметрів, граничні умови. При натисканні на кнопку «сформувати» в системі генерується розрахункова різницева схема з заданим законом розподілу та з відповідними розрахунковими параметрами. Розмірність системи відповідає вказаній кількості вузлів за просторовими координатами.

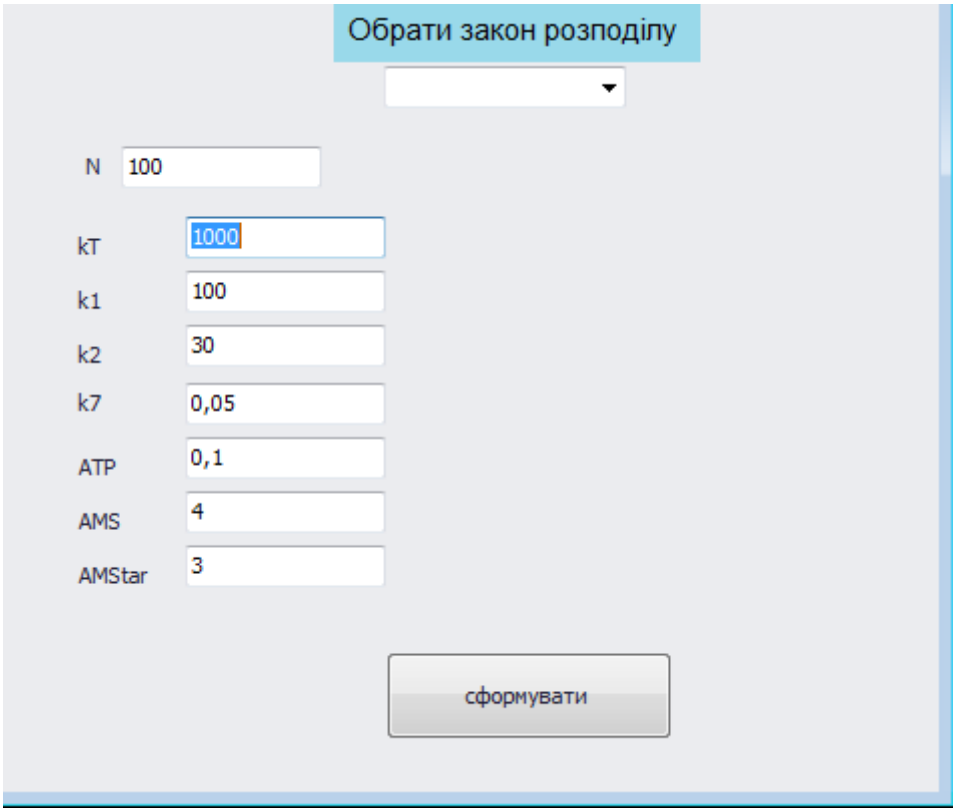


Рисунок 4 - Зовнішній вигляд форми формуваня даних

4. ТЕСТОВА РЕАЛІЗАЦІЯ МОДЕЛІ У РОЗРОБЛЕНОМУ ПРОГРАМНОМУ СЕРЕДОВИЩІ

При моделюванні процесів взаємодії в розробленій програмній системі для дослідження було виділено 3 основні параметри, серед яких: загальна швидкість реакції взаємодії, граничне значення сили, розмір вікна спостереження реакції. Під час зміни швидкості гідролізу АТФ (аденозітріфосфорна кислота) поступово змінювалися значення швидкості реакції на кожній із 6 зазначених стадій. Результати моделювання наведено у таблиці 1.

Таблиця 1

Чутливість результатів моделі до зміни параметрів

Параметр	Зміна	Значення сили напруги	Розмір вікна спостереження
Реакція 1 – k1	+	+	+
	-	-	-
Реакція 2 – k2	+	+	+
	-	-	-
Реакція 5 – k5	+	Незмінно	Незмінно
	-	Незмінно	Незмінно
Реакція 7 – k7	+	-	+
	-	+	-

Розроблена програмна модель дозволила підтвердити результати взаємодії, отримані експериментальним шляхом. Завдяки використанню засобів моделювання, були отримані результати, що підтверджують припущення про те, що взаємодія при номінальних параметрах можлива тільки при використанні концентрації АТФ в межах від 0.1 до 10. У тих випадках, коли концентрація занадто велика або мала, відбувається гідроліз АТФ, що тягне за собою відсутність реакції.При комп'ютерному моделюванні були внесені істотні зміни в діапазони варіації параметрів і передбачена можливість розриву ниток актину під впливом загальної сили. В результаті моделювання визначено межі інтервалів концентрації кислот, що забезпечують взаємодію між молекулами міозину і нитками актину. Робота програми підтверджує гіпотези, що було висунуто в [2]. Найбільш вагомий результат полягає у побудові графіку залежності між концентрацією АТФ у клітинах та значенням сили, що виникає під час реакції рис. 5

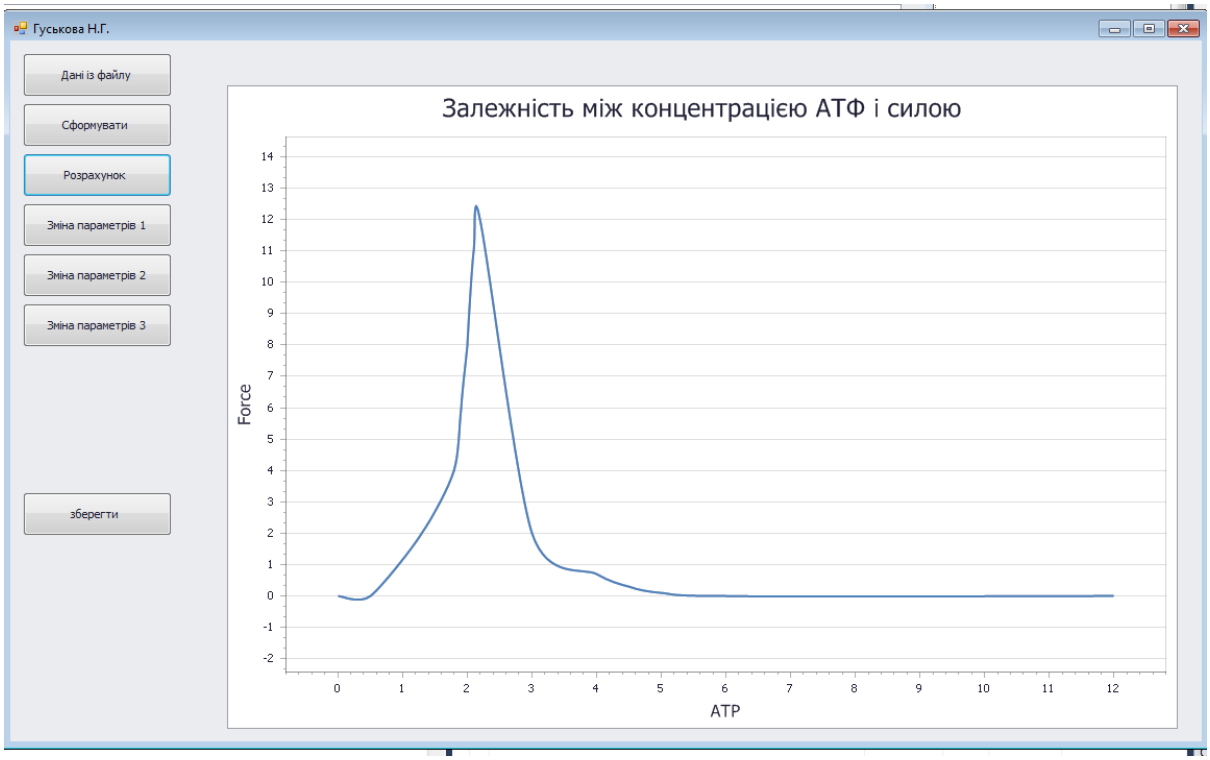


Рисунок 5.22 - Залежність між концентрацією АТФ і силою

5. ВИСНОВКИ

Матеріал, наведений у роботі, характеризує результати проведених досліджень, пов'язаних розробкою програмної системи для моделювання взаємодії білків актину і міозину в процесі скорочення клітини. Визначено основні вимоги щодо проектування програмної системи. Розроблено структуру програмної системи, яка складається з декількох функціональних блоків. В програмній системі окремо виділено блоки, що забезпечують формування початкових даних, задають початкові і граничні умови відповідно до обраного типу системи, що моделюється. Передбачено функціональний блок, який дозволяє обрати явну або неявну схему дискретизації з відповідними додатковими умовами, що пов'язані з точністю апроксимації, виконанням обмежень на крок інтегрування, початковими і граничними умовами, заданими параметрами моделі. Для обробки результатів передбачено блок візуалізації, де чисельні розрахунки представляються у вигляді графіків та таблиць параметрів.

Опис структурних модулів програмної системи здійснено за допомогою UML діаграм, а саме: діаграми варіантів використання, діаграми класів, діаграма послідовностей підпрограми вводу початкових даних, діаграма послідовностей підпрограми розрахунків.

Досліджено розподілену математичну модель взаємодії і визначено наявності залежностей між концентрацією кислот і силою напруги, яка виникає в моменти стиснення клітини. Ґрунтуючись на модифікації базової моделі було визначено взаємозв'язки для обчислення концентрації речовини.

Проведено дослідження чутливості параметрів для зосередженої моделі. На основі визначення сили напруги, здійснено вибір оптимального числа точок та оптимальної концентрації АТФ. Обґрунтовано обрані схеми дискретизації рівнянь, виконано стабілізацію системи.

Отримані залежності для оцінки чутливості результату до зміни кінетичних параметрів моделі підтверджуються результатами лабораторних експериментів.

ЛІТЕРАТУРА

[1] Святный В.А. Разработка и реализация математической модели стабилизации взаимодействия белков актина и миозина/ В.А. Святный, Н.Г. Гуськова // Наукові праці ДонНТУ. Серія "Інформатика, кібернетика та обчислювальна техніка" (ІКОТ-2016). – 2016. – №1(22). – С. 146-151

[2] Святный В.А. Оптимизация параметров распределенной модели массового взаимодействия/ В.А. Святный, Н.Г. Гуськова // Наукові праці ДонНТУ. Серія "Інформатика, кібернетика та обчислювальна техніка" (ІКОТ-2016). – 2016. – №2(23). – С. 110-114

[3] Lewis, O. Actin-myosin spatial patterns from a simplified isotropic viscoelastic model / O. Lewis, R. Guy, J. Allard // Biophysical Journal. – 2014. – Vol. 107, №4. – P. 863-870.

[4] Vogel S. Myosin motors fragment and compact membrane-bound actin filaments / S. Vogel, Z.Petrasek, F.Heinemann// ELife. – 2013.Vol.2 e00116. P. 1-18.

[5] Gou J. Mathematical model with spatially uniform regulation explains long-range bidirectional transport of early endosomes in fungal hyphae / J. Gou, L. Edelstein-Keshet, J. Allard// Molecular Biology. – 2014. – Vol. 25, №16. – P. 2408-2415.

[6] Самарский А.А. Численные методы / А. А. Самарский, А. В. Гулин. – М.: Наука, 1989. – 432с. – ISBN 5-02-013996-3.

[7] Дмитриева О.А. Высокоэффективные алгоритмы управления шагом на основе параллельных коллокационных блочных методов // Искусственный интеллект. – 2012. – № 4. – С. 77 – 88.

Отримано 20.06.2017

УДК 004.272.2:519.63

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ПРОГРАМНОЇ СИСТЕМИ ПАРАЛЕЛЬНОГО МОДЕЛЮВАННЯ ДИНАМІЧНИХ ОБ'ЄКТІВ

О.А. Дмитрієва, Т.Г. Дараган
Донецький національний технічний університет, м. Покровськ

Роботу присвячено питанням розробки та тестування програмної системи паралельного моделювання складних динамічних об'єктів, що описуються системами звичайних диференційних рівнянь (СЗДР). Проведено дослідження ефективності паралельних блокових методів з автоматичним керуванням кроку, запропоновано оптимізаційні схеми комунікаційних операцій.

Ключові слова: паралельне моделювання, блокові методи, комунікаційні операції, керування кроком, MPI

1. ВСТУП

На теперішній час одним з найбільш поширених і ефективних методів дослідження динамічних систем є комп'ютерне моделювання, яке відзначається широким застосуванням паралельних обчислювальних систем різноманітних архітектур, топологічних характеристик, програмних інструкцій. При цьому виникає потреба в ефективних паралельних методах розв'язання задачі Коші для систем звичайних диференціальних рівнянь, які використовуються при моделюванні динамічних об'єктів з зосередженими параметрами. В роботі в якості таких методів використовуються блокові колокаційні методи, які характеризуються абсолютною стійкістю [1] та збіжністю за правою частиною і за початковими даними [2], що дозволяє використовувати їх при вирішенні жорстких задач, але разом з цим зростає складність побудови на їх основі алгоритмів автоматичної зміни розміру кроку інтегрування [3]. Колокаційні методи для розв'язання задачі Коші будуються на інтерполяційних многочленах [4], причому їх ступені відповідають кількості точок колокації, а значення многочленів в цих точках збігаються з правими частинами диференціальних рівнянь в розрахункових точках

Для вирішення поставленої задачі було спроектовано програмну систему, яка складається з двох підсистем, де перша включає реалізацію послідовних, паралельних та модифікованих паралельних алгоритмів розв'язання СЗДР блоковими методами, множину тестових завдань для проведення обчислювальних експериментів.

Друга підсистема реалізує візуалізацію отриманих результатів. Розробку програмної системи виконано згідно методології об'єктно-орієнтованого програмування з використанням сучасних програмних засобів. Підсистему для проведення паралельних обчислень ParallelBlock було написано з використанням принципів об'єктно-орієнтованого програмування для платформи Microsoft .NET на мові C# з використанням інструкцій MPI.

2. ОПИС СТРУКТУРИ ПРОГРАМНОЇ СИСТЕМИ

Розроблена програмна система поділяється на дві окремі підсистеми: розрахункову, та підсистему візуалізації, яка детально відображує результати розрахунків. Запустивши першу підпрограму, яка відповідає за обчислення, необхідно обрати кількість процесів, що будуть задіяні у роботі програми. Програма може буде запущена в розподіленій системі з середовищем MPI на необхідній кількості процесорів. Структурна схема програмної системи зображена на рис. 1 і містить наступні модулі:

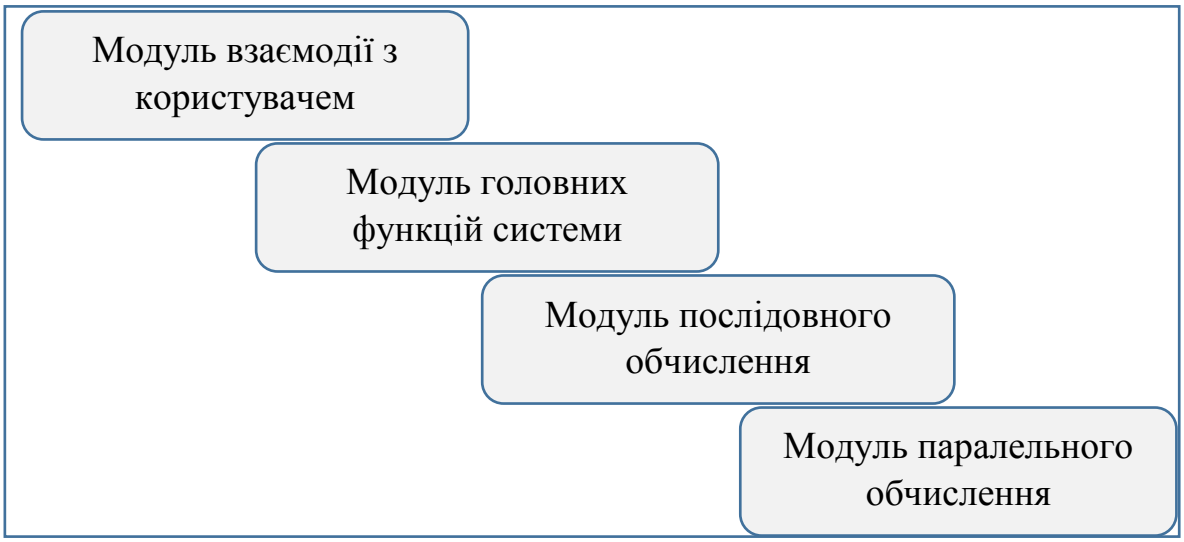


Рисунок 1 – Схема модулів програмної системи

1) головний модуль, який відповідає за збір початкових даних від користувача та запуск обраної задачі на заданому числі процесорів;
2) модуль функцій, який включає головні функції програмної системи, а саме: генерацію початкових стадійних коефіцієнтів, обчислення кроку, обмін повідомленнями між процесорами;
3) модуль з послідовною реалізацією блокових методів для розв'язання систем диференційних рівнянь для тестових задач;

4) модуль з паралельною реалізацією блокових методів.

Після закінчення розв'язання задачі дані результатів зберігаються у вихідний файл, тому користувач має можливість переглянути отримані дані, запустивши для цього другу підпрограму системи. Для перегляду результатів обчислення необхідно обрати файл з результатами та відкрити їх у програмі. Вихідний файл містить результати обчислень для детального аналізу: кількість задіяних процесорів, кількість відкинутих кроків, кількість прийнятих кроків, результати обчислень у вигляді одновимірного масиву.

Друга підсистема складається з одного модулю, головна функція якого полягає у зчитуванні результатів з файлу та відображенні їх в графічному вигляді. На рис. 2 зображена діаграма класів програмної системи, що спроектовано.

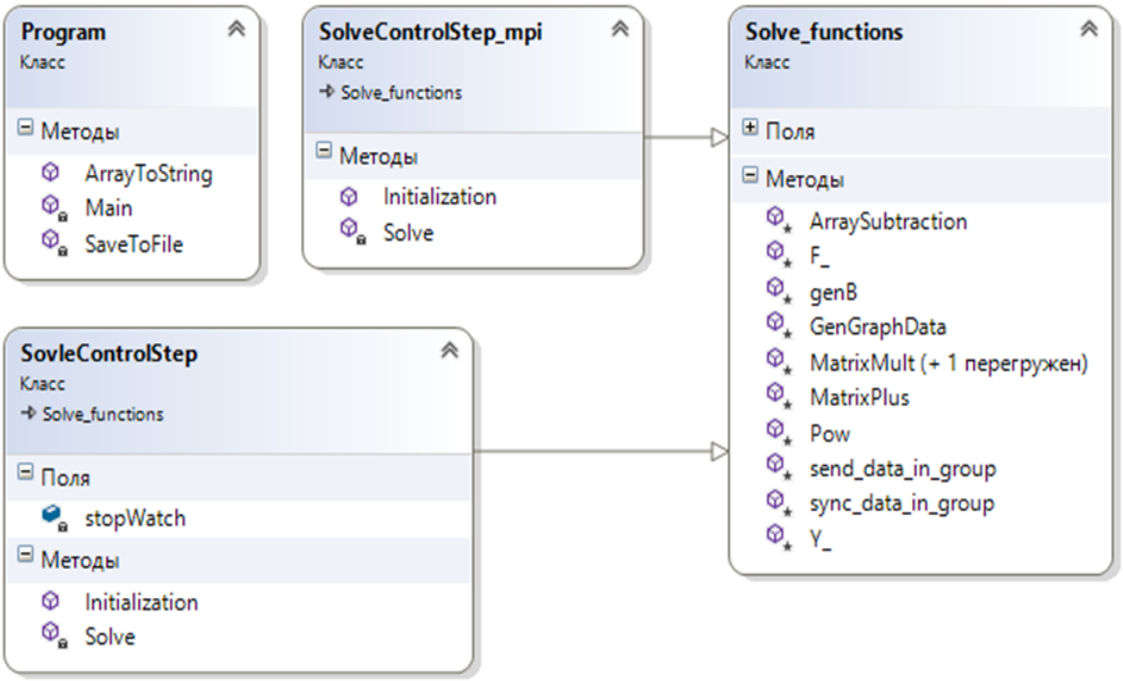


Рисунок 2 – Діаграма класів

Головний клас Program, який виконується першим при запуску підсистеми для обчислення, отримує початкові данні від користувача, передає ці данні до необхідного класу, в залежності від обраного методу розв'язання задачі, та запускає процес обчислення. Якщо було обрано послідовний алгоритм обчислення, то данні передаються до класу SolveControlStep, при обранні паралельного алгоритму обчислення данні передаються до класу SolveControlStep_mpi.

У класі Solve_function об'явлені атрибути (поля), які наслідуються класами SolveControlStep та SolveControlStep_mpi.

Також клас Solve_function містить методи, які є загальними як для послідовного, так і для паралельного алгоритмів обчислення, лише з однією різницею, яка полягає у тому, що методи send_data_in_group() та sync_data_in_group() використовуються у паралельному алгоритмі для передачі повідомлень між задіяними процесами. Класи SolveControlStep та SolveControlStep_mpi відрізняються також тим, що для ініціалізації розрахунку аналогічні методи приймають різну кількість аргументів, тому що для паралельного обчислення необхідні додаткові початкові данні, які стосуються кількості процесорів.

3. АНАЛІЗ ЕФЕКТИВНОСТІ МОДЕЛЮВАННЯ ДЛЯ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ

При проведені тестування блоковими методами обчислювалась задача Капса [5] виду

$$x_1'(t) = -(\lambda + 2)x_1(t) + \lambda x_2^2(t),\ \ x_1(0) = 1,\ \ 0 \le t \le 10,$$
$$x_2'(t) = x_1(t) - x_2(t)\big(1 + x_2(t)\big),\ \ x_2(0) = 1, \qquad (1)$$

з відомими точними розв'язаннями:

$$x_1(t) = e^{-2t},\ \ x_2(t) = e^{-t}. \qquad (2)$$

До початку проведення обчислень було згенеровано розрахункові коефіцієнти для опорних та розрахункових блоків. Також для запуску процесу обчислень було застосовано варіативні значення коефіцієнту жорсткості λ, початкового кроку інтегрування τ_1, коефіцієнту трудомісткості реалізації правих частин (коефіцієнту навантаження).

Тестування програмної системи з паралельним алгоритмом розв'язання задачі Коші для СЗДР виконувалося на комп'ютері з 4 доступними процесами, тому результати обчислень, а саме, час розрахунку тих задач, де число заданих процесів дорівнює 6 та 8, значно вище, ніж при обчисленні 2 або 4 процесами. Це пояснюється тим, що операційна система додатково розподіляє ті обчислення, які були закріплені за 5, 6 та іншими процесами, на доступні 4, використовуючи потоки. Однак отримані результати все одно дозволяють оцінити ефективність отриманих результатів обчислення.

На рис. 3-4 відображено результати обчислень системи рівнянь (1) з коефіцієнтами жорсткості λ=1 та λ =100, та коефіцієнтом навантаження 5000. Порівнюючи отримані результати, видно, що при обчисленні системи рівнянь з λ=1 не було відкинутих кроків, а навпаки, крок інтегрування постійно зростав. Однак при λ=100 з'явились відкинуті кроки інтегрування, та були спроби їх зменшення,

що означає збільшення загальної кількісті ітерацій при обчисленні кроків.

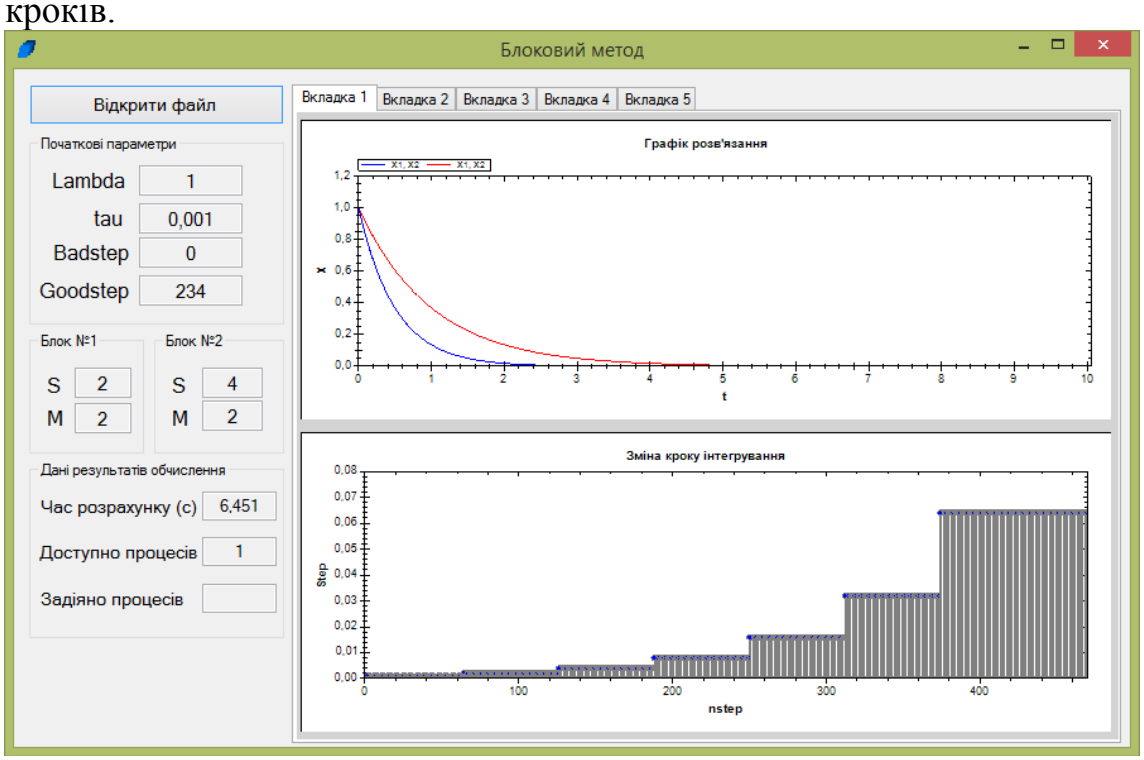


Рисунок 3 – Результати обчислення системи (1) при λ=1

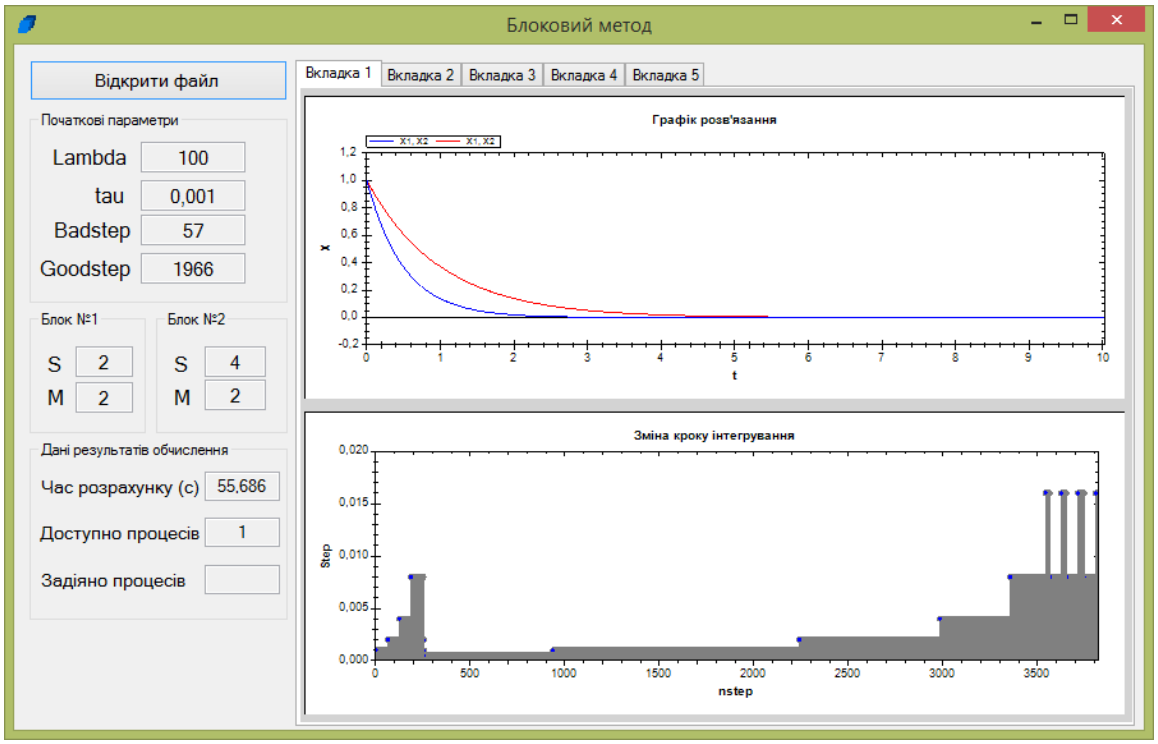


Рисунок 4 – Результати обчислення системи (1) при λ=100

Графік залежності часу обчислення від трудомісткості реалізації правих частин на 1, 2 та 4 процесах зображено на рис.5.

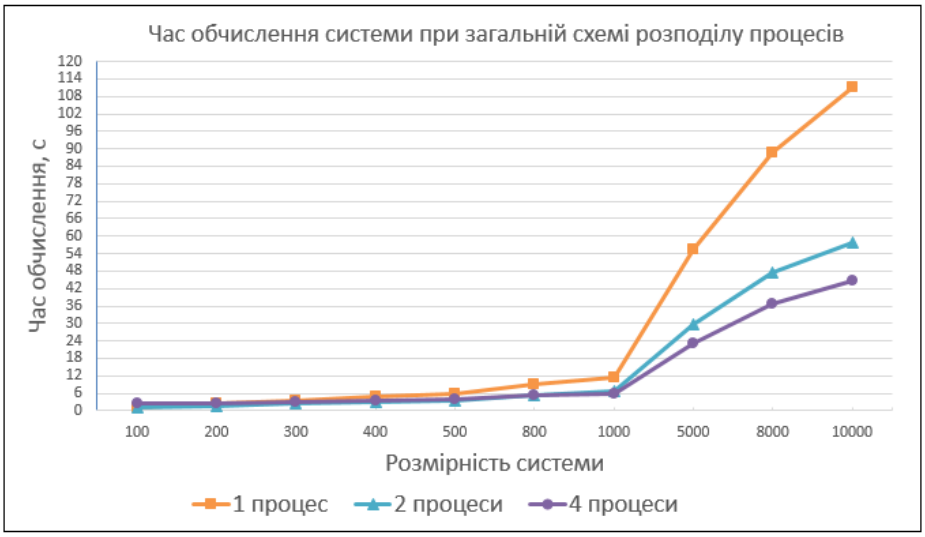


Рисунок 5 – Графік залежності часу обчислення від розмірності системи (1) на 1, 2 та 4 процесах

Графіки локальної похибки при обчисленні системи рівнянь на 4 процесах зображено на рис. 6, який відображає спроби збільшення або зменшення кроку інтегрування на всьому відрізку.

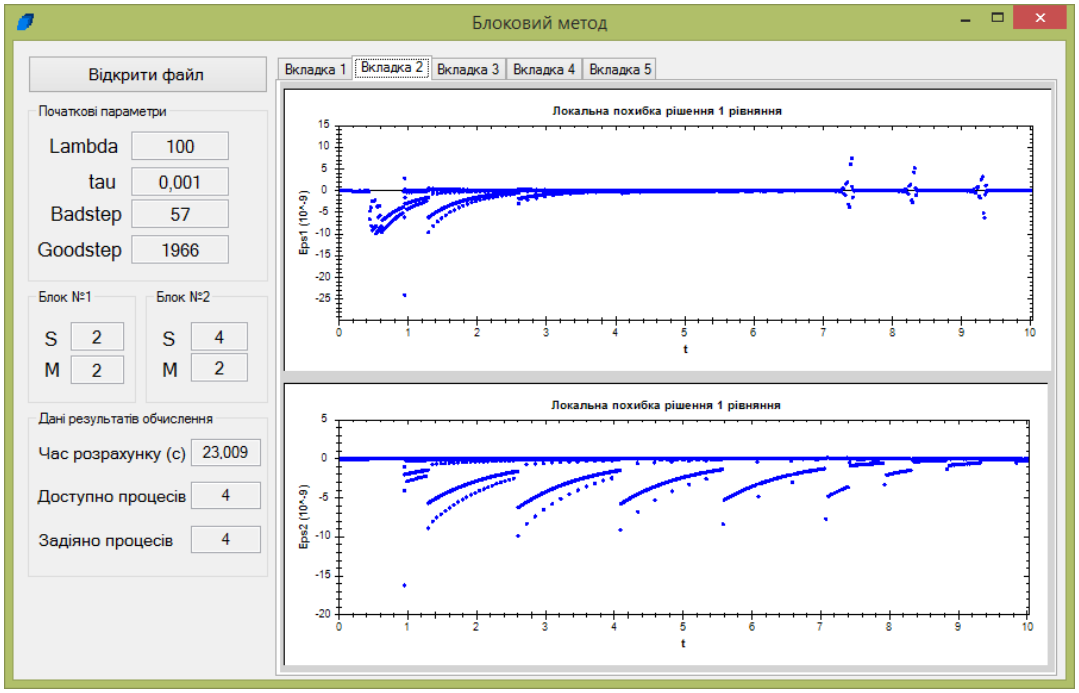


Рисунок 6 – Графіки локальної похибки для рівнянь системи (1)

Таким чином, запропонований та досліджений метод оптимізації комунікаційних операцій керуванням кроку в паралельних алгоритмах моделювання мінімізує обчислювальні ресурси при вирішенні систем диференційних рівнянь.

4. ВИСНОВКИ

В роботі розглянуто сучасні підходи до чисельного розв'язання задачі Коші. Проведено дослідження ефективності паралельних блокових методів з автоматичним керуванням кроку. Розглянуто проблему вибору ефективної кількості обчислювальних вузлів для паралельного розв'язання блоковими методами. Розроблено оптимізаційні схеми комунікаційних операцій при керуванні кроком інтегрування. Проектування програмної системи здійснювалося на підставі засобів візуального моделювання з використанням діаграм UML. Реалізовано програмну систему, що включає реалізацію послідовних, паралельних та модифікованих паралельних алгоритмів розв'язання СЗДР блоковими методами, множину тестових завдань для проведення обчислювальних експериментів. Розробку програмної системи виконано згідно методології об'єктно-орієнтованого програмування з використанням сучасних програмних засобів з використанням інструкцій MPI

ЛІТЕРАТУРА

[1] **Дмитриева О.А.** Коллокационные блочные методы с контролем на шаге/О. А. Дмитриева //Системи обробки інформації. – 2015, № 5(130). – С. 78-84.

[2] **Дмитриева О.А.** О построении параллельных разностных схем моделирования с вариацией шага в расчетном блоке /Дмитриева О.А // Вестник НТУ "ХПИ". Серия: ИМ. – 2016. – № 21(1193). – С. 20 - 28.

[3] **Дмитриева О.А.** Параллельные численные методы моделирования динамических объектов: монография / О.А. Дмитриева. – Красноармейск: ГВУЗ «ДонНТУ», 2016. – 384 с

[4] **Dmitrieva O.** Parallel Step Control. Development of parallel algorithms of the step variation for simulation of stiff dynamic systems/ O. Dmitrieva, L. Feldman. - Lambert Academic Publishing. – 2013. – 72 p.

[5] Numerical Solution of Extended Block Backward Differentiation Formulae for Solving Stiff ODEs / [S.A.M. Yatim, Z.B. Ibrahim, K.I. Othman, M.B. Suleiman] //Proceedings of the World Congress on Engineering. WCE 2012, July 4 – 6, 2012, London, U.K. – 2012. – Vol. 1. – P. 109–113.

Отримано 20.06.2017

УДК 004.272.2:519.63

ДОСЛІДЖЕННЯ ЕКСТРАПОЛЯЦІЙНОГО МЕТОДУ ДОРМАНА-ПРИНСА 5(4) З КЕРУВАННЯМ КРОКУ

О. С. Бобилєва, О. А. Дмитрієва
Донецький національний технічний університет, м. Покровськ, Україна

Розглянуто паралельну реалізацію екстраполяційного методу Дормана-Принса 5(4) з керуванням кроку та проведено ряд програмних експериментів. Результати роботи наведено у таблиці та на графіках, які допомагають проаналізувати отримані показники.

Ключові слова: задача Коші, вкладені методи, метод Дормана - Принса, керування кроком.

На сьогоднішній день дуже актуальною є задача математичного моделювання динамічних процесів, що описуються системами звичайних диференційних рівнянь (СЗДР), а саме пошук розв'язання задачі Коші за допомогою розробки і реалізації чисельних методів [1-2]. Як правило, динамічні об'єкти, до яких відносять фізичні тіла і системи зв'язаних тіл, явищ, а також технологічні процеси, які можуть сприймати зовнішні фізичні впливи і відгукуватися на них зміною вихідних фізичних величин, що характеризують їх стан і поведінку, описують диференціальними рівняннями. Різноманіття видів динамічних об'єктів визначає високу значимість диференціальних рівнянь як математичного апарату їх опису, що дозволяє проводити теоретичні дослідження цих об'єктів і на основі такого аналізу конструювати моделі і будувати системи [2].

В даний час методи типу Рунге – Кутти є найбільш використовуваними однокроковими методами вирішення чисельного розв'язання СЗДР. В якості методу дослідження було обрано один з методів, які відносяться до вкладених методів типу Рунге-Кутти - метод Дормана-Принса. Для досягнення поставленої мети було використано методологію об'єктно – орієнтованого програмування та стандарт MPI. Тестування розробленої програмної системи виконано на багатоядерному комп'ютері при проведенні чисельних експериментів.

Для проведення досліджень було розроблено програмну систему, що містить в собі паралельну реалізацію чисельного методу

розв'язання СЗДР Дормана – Принса 5(4). Система реалізована у вигляді консольного додатку операційної системи.

З метою дослідження ефективності паралельної реалізації методу Дормана – Принса для систем з розподіленою пам'яттю проведено ряд чисельних експериментів, що спрямовані на розрахунок системи рівнянь Бутчера з розмірністю системи 4, 10, 20 і 30 рівнянь, та точністю розрахунку 10^{-9} з використанням 1, 2 та 4 процесорів, та параметрів методу *facmin*=0,5, *facmax*=1,5, та початковим кроком інтегрування 0,001 [3].

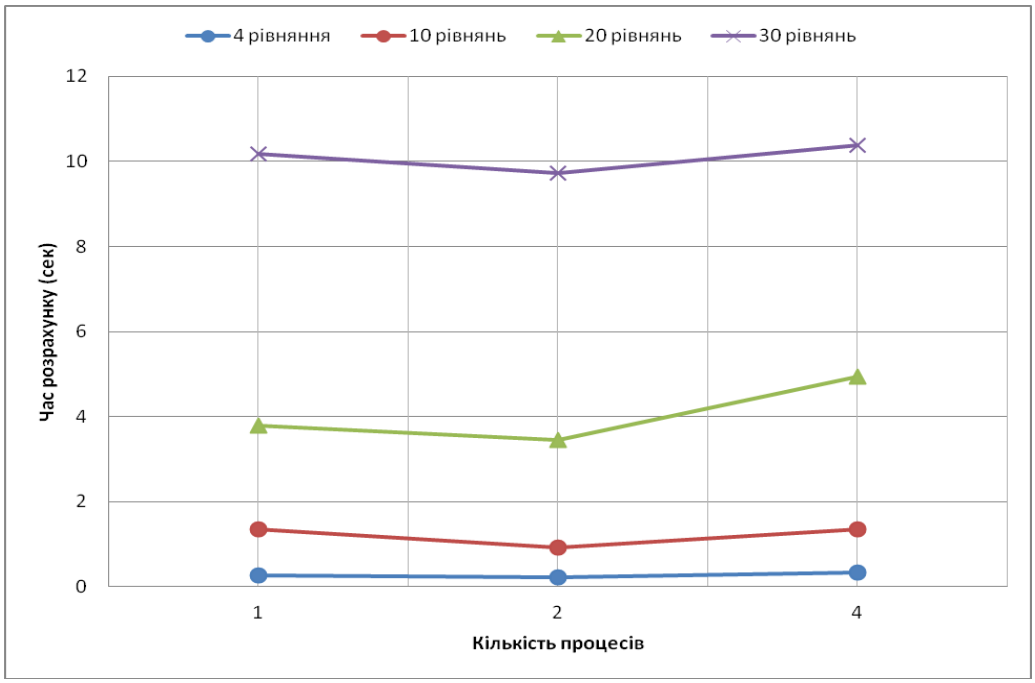


Рисунок 1 – Графік залежності часу розрахунку від кількості задіяних процесів

Таблиця 1 – Розрахунок системи Бутчера з 4, 10, 20 і 30 рівняннями паралельним методом Дормана – Принса 5(4)

Кількість задіяних потоків	Час виконання методу Дормана – Принса 5(4)(сек)			
	Система Бутчера з 4 рівняннями	Система Бутчера з 10 рівняннями	Система Бутчера з 20 рівняннями	Система Бутчера з 30 рівняннями
1	0,258	1,357	3,795	10,177
2	0,224	0,91	3,452	9,732
4	0,326	1,353	4,942	10,384

Експерименти проводилися з використанням реалізованого методу Дормана – Принса 5(4). У зв'язку з відсутністю доступу до реальної розподіленої обчислювальної системи експерименти проводилися на комп'ютері з багатоядерним процесором за допомогою *mpiexec*. Команда mpiexec дозволяє запускати паралельні програми на необхідній кількості процесів. Результати експериментів відображено на рис. 1 та наведено у табл. 1.

Як видно з рис. 1, на 2 процесах програма працює з меншим часом розрахунку ніж на 1, але на 4 процесах час зростає. В якості можливих причин відхилення від максимального прискорення можна назвати:

– велика кількість операцій обміну, конфлікти пам'яті;

– прискорення паралельного обчислення менше часу, що потрібен на виконання повільних операцій;

– недостатня пам'ять;

– незбалансованість навантаження процесорів [4].

Проведені дослідження дозволили запропонувати нові підходи до вирішення проблеми паралельного управління точністю інтегрування на основі варіації довжини кроку.

В подальшому робота отримує інтерфейсну реалізацію, будуть досліджені можливості підвищення прискорення методу Дормана – Принса 5(4) та розроблено реалізацію для кластерної системи. Результати роботи можуть бути використані як демонстраційний матеріал для студентів, що навчаються за напрямком «Інженерія програмного забезпечення» та за суміжними напрямками.

ЛІТЕРАТУРА

1. **Хайрер Э.** Решение обыкновенных дифференциальных уравнений. Нежесткие задачи / Э. Хайрер, С. Нерсетт, Г. Ваннер – М.: Мир, 1990. – 512с.

2. **Хайрер Э.** Решение обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи // Э. Хайрер, Г. Ваннер – М.: Мир, 1999. – 685с.

3. **Дмитриева О. А.** Разработка параллельных алгоритмов управления шагом на основе вложенных стадийных методов / О. А Дмитриева // Наукові праці Донецького національного технічного університету, серія: «Проблеми моделювання та автоматизації проектування» № 1 (10)-2(11), 2012 с. 22-30.

4. Динамические объекты Определения, модели, структура, свойства [Електронний ресурс]. – Режим доступу: http://model.exponenta.ru/bt/bt_00414_Din_Obj_1.htm

Отримано 20.06.2017

УДК 004.9:622.451

FUZZY CONTROL ALGORITHM SIMULATION

S. Kovalov

Donetsk National Technical University, Ukraine.

The article is devoted to the questions of the control algorithm simulation for the coal mine excavation site ventilation using fuzzy logic. Such an algorithm allows the use of expert assessments to improve the quality and safety of air management, which is important for modern coal mining technologies. The features of fuzzy algorithm simulation in MATLAB Simulink environment are considered.
Key words: fuzzy control system, linguistic variables, fuzzy inference, coal mines excavation site, ventilation, transient processes by air and methane.

Introduction

A fuzzy control system is a control system based on fuzzy logic – a mathematical system that represents analog input values in terms of linguistic variables. These variables take on continuous values between 0 and 1 in opposite to classical or digital logic, which operates on discrete values of either 1 or 0 (true or false, respectively). Fuzzy techniques have been successfully used in control. Engineers and Researchers are today considering fuzzy logic algorithms in order to implement intelligent functions in control systems [1,2].

Permanently working information system for parameters and components of the mine atmosphere monitoring should be put in coal mines to ensure the safety of coal mining procedure. The mine atmosphere's main components are air and methane [3]. Using data from such a system, it's possible to automate of ventilation management processes to improve their safety and effectiveness. This is important in connection with the increase of the load on the mine site and the increase of the mines' depth. Thus there is a need to establish a control system of mine's atmosphere that would provide not only the uninterrupted monitoring of information, but also the impact on the object of control to align the attributes and components of ventilation flow corresponding with the requirements of the safety rules.

A lot of research has shown that changing ventilation's mode of the coal mines excavation site (CMES), which are hazardous on gas factor, may lead to an undesirable excess of the allowable values of methane concentration during the transition process [3,4]. Therefore, algorithms for ventilation control should be safe not only in stationary mode, but also

methane concentration should not exceed the safe level during transient processes by air and methane (TPAM).

Organization of the fuzzy control system model

The model of the fuzzy control system of coal mine excavation site ventilation has been developed for research these safe algorithms. The model has been developed in MATLAB simulation environment [5]. The model has a block structure, whose main elements are:

block of transient processes simulation for CMES;

block of the actuators (air flow regulators) models;

model of the fuzzy control algorithm of airing.

The following scheme of assumptions and interactions between the variables of the model's components was used during the research.

Block of transient processes simulation for CMES. It allows calculate changes of the methane concentration in CMES depending on the parameters of the ventilation circuit or air depression changes on the input poles of the site. The model for the research of the control algorithm is based on mathematical description of TPAM and parameters of ventilation schemes that have been used in [4, 6]. Structure of the MATLAB model is shown in the figure 1. It includes 3 blocks: Q-system, Q_m-system and C-system. First block is designed for simulation of air distribution on the branches of CMES. These values of air flow are used for formation of the methane flow from worked out area of the excavation site Q_m. in the block Q_m-system. Air and methane flows are input variables for C-system model block, which provides simulation of the methane concentration c_i transient process on CMES.

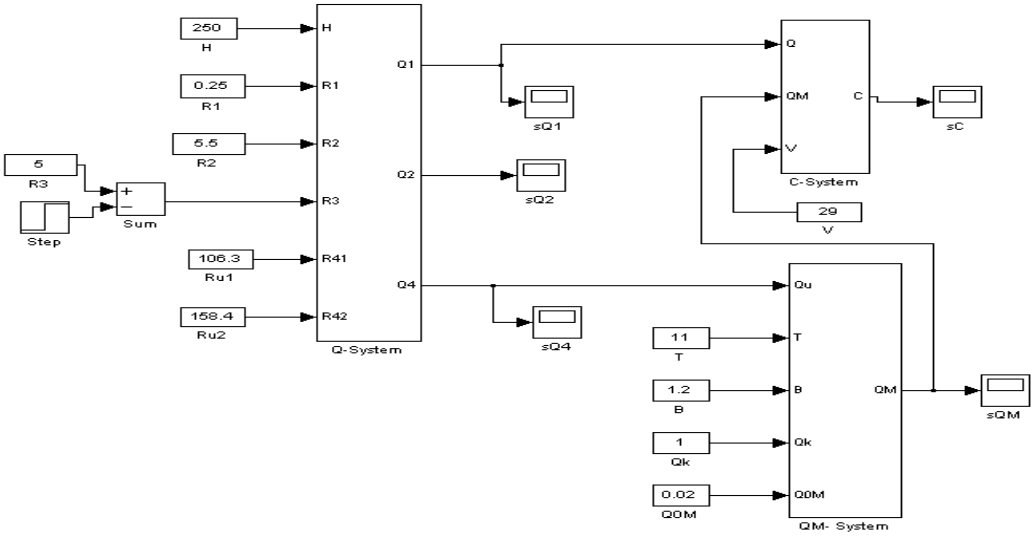


Figure 1. MATLAB model structure of the control object - the ventilation circuit of CMES.

From point of view of the control system the model is subject to control, where the output variable is the air flow controller installed into the ventilation drift of excavation site. The input variables are methane concentrations and air flow of the ventilation drift. Parameters of the ventilation circuit for 7th northern face of coal mine "Kholodnaya Balka" №3 were used in the model. The identification of MATLAB model parameters were carried out by comparison of gas dynamics plot in a real experiment and model. The results of TPAM simulation show sufficient accuracy of the selected model parameters (standard deviation doesn't exceed 9.65%) [6].

Model of actuator controls of the air flow. It's a model of flow rate regulators. Usually, it's the object with non-linear transfer function. It can be taken into account for membership function of output variable in a control system based on fuzzy logic. Furthermore, the time constant of control is sufficiently smaller than a time constant of methane transients. All of this allows reducing the model of actuator to the adjustable resistance of site output branch which is calculated for the accomplishment of required air flow rates.

Model of fuzzy control algorithm of airing. Fuzzy control algorithm model implemented by A. Mamdani approach and includes the steps of fuzzification, defuzzification and logical processing [7]. At the stage of fuzzification in the system variables are defined on the basis of which a decision is made on the control action, and they are produced by the membership functions of the linguistic variable values. Logical processing uses the rules of the output variable formation for different combinations of input actions on the basis of an expert assessment

Consider of the model elements formation for the control system taking into account the control object features. In accordance with the safety of mining the concentration of methane in mines is limited to a certain level. So, violation of the safety for some excavation sites on the gas ratio may arise in the event of the need to reduce the concentration of methane. This is confirmed by the simulation results for the conditions of the ventilation circuit of 7th northern face of mine "Kholodnaya Balka" №3.The methane concentration surge appears during changing the methane concentration from 0.4% to 0.33%. The amplitude of surge comes to 0.56% and the transition process for gas continuous more than 3 hours [6]. Perturbing influence here is 21% step increase in air flow rate. Assuming that it is needed to perform similar change of the ventilation mode with limited level of methane concentration should be used the special algorithm to modify the air flow, while controlling the methane concentration deviation from the permissible level and its rate of change.

The base of rules for the fuzzy control system

To ensure safe transfer of the ventilation circuit of CMES into a new ventilation mode can be used system with fuzzy control algorithms [1,7]. Such system requires a framework of rules that use linguistic variables and their values are formed at the stage of fuzzification using membership functions. For safe reducing of current methane concentration c_i expert can estimate the value of its deviation from safe level and speed of c_i changing. Needed value of air flow changing is possible to apply as output linguistic variable.

So, Two-Input Single-Output Mamdani fuzzy inference system [7] can be used the rules of the following form:

- ***IF*** deviation c_i from the permissible level of methane concentration <value>
- ***AND*** the rate of the methane concentration change <value>
- ***THEN*** the change of the air flow <value>.

This is two input linguistic variables – «the deviation c_i from the permissible methane concentration levels» (with 4 value terms) and «the rate of the methane concentration change» (with 5 value terms). Output variable "the air flow» has following value terms: "strong increase", "increase", "not change", "decrease", "strong decrease". Table 1 summarizes values of the rules for the safe changing of the methane concentration using represented linguistic variables.

Table 1. The base rules values of the fuzzy control system

The concentration of methane	The deviation c_i from the permissible level is			
	very small	small	middle	big
quickly increases	**strong decrease**	**decrease**	**not change**	**not change**
increases	**decrease**	**not change**	**not change**	**increase**
not changing	**not change**	**increase**	**increase**	**strong increase**
decreases	**not change**	**increase**	**increase**	**strong increase**
quickly decreases	**not change**	**increase**	**strong increase**	**strong increase**

It includes information for 20 IF-THEN rules. Each rules has same form and value of the output variable indicates into the inside cells of the table (bold). Example of 20th rule (output value into the bottom right corner cell) is: ***IF*** deviation c_i from the permissible level of methane concentration **big** ***AND*** the methane concentration **quickly decreases** ***THEN*** the air flow **strong increase**.

The membership functions (MF) were developed for linguistic variables using MF Editor for SIMULINK Fuzzy Program [5]. For Two-Input Single-Output Mamdani fuzzy inference system it was such MF organization:
Input A: half-trapezoidal, full-trapezoidal, full-trapezoidal, half-trapezoidal MFs (100% OR);
Input B: half-trapezoidal, full-trapezoidal, full-triangle, full-trapezoidal half-trapezoidal MFs (100% OR);
Output: half-triangle, full-triangle, full-triangle, full-triangle, half-triangle MFs (100% OR).

The rate of change of the concentration of methane (Input B) was determined as the ratio of the difference of the methane concentration sensor in the current and previous polling cycle for the duration of the polling interval. Sensitive of the system for this parameter was 0,002% per minute. Big deviation c_i from the permissible level (Input A) was more then 0,07%. The value of the air flow «increase» (max) is 5%, the air flow «strong increase» (max) is 10%. The output linguistic variable on the defuzzification step is transformed into the control action as the middle of the maximum active region of the output variable after inference [7].

Figure 1 show the example of the transient response on air and methane to reduce methane concentration from 0.4% to 0.33%, where the restriction's level is installed up to 0.5%.

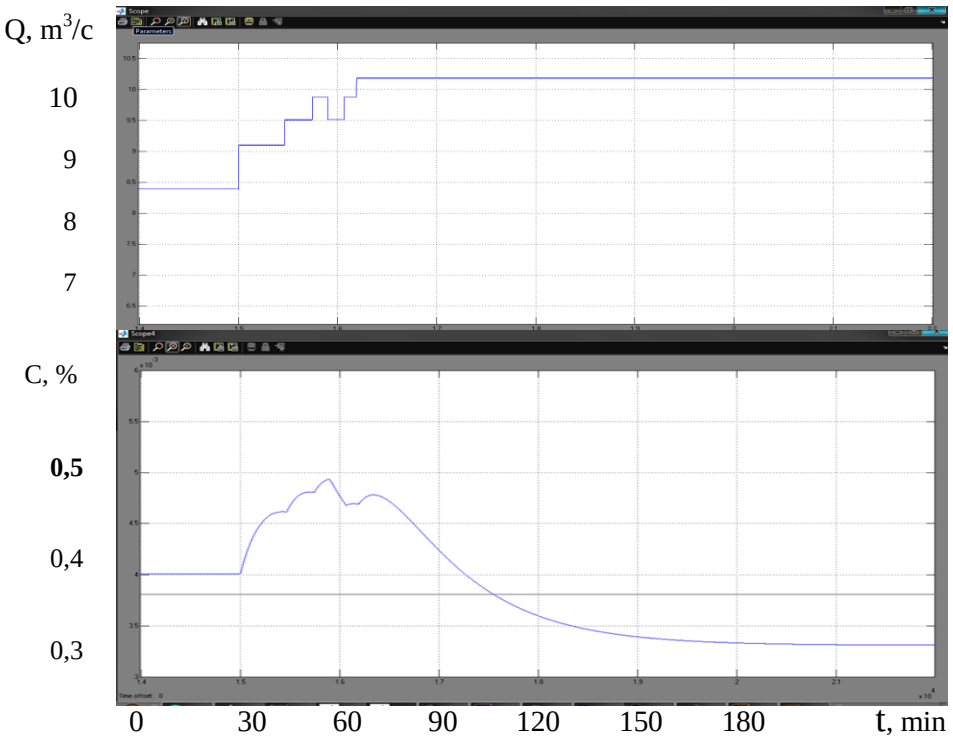


Figure 1 – The transient response on air and methane

The simulation was performed in MATLAB using developed model of the control system. The sensors' polling interval for this experiment was 5 minutes. Taking into account the gas dynamics process in each step of changing the mode of ventilation, the fuzzy control algorithm provided to limit the methane concentration at the desired level (0.5%). It was made research variants of the fuzzy algorithm working for different polling intervals. All simulation experiments are shown, that designed fuzzy algorithm of control provides changing of methane concentration with the limited level for any polling intervals. It was analyzed length of TPAM and quantity steps of control during this process Obtained as a result of simulation the sensors polling interval (3-5 minutes) allows decrease of the transient process time using minimum quantity of the regulation cycles.

Conclusions

A model of fuzzy control systems was developed in MATLAB Simulink environment. Research of the transient processes by air and methane allowed developing the base of rules for Two-Input Single-Output Mamdani fuzzy inference system, to specify the membership functions of linguistic variables and demonstrate the effectiveness of selected rules for different modes of ventilation and polling intervals of sensors in the fuzzy control system.

References

[1] KevinM.Passino, StephenYurkovich. Fuzzy control - Addison Wesley Longman, Inc., 1998, 502 p. ISBN0-201-18074-X

[2] Zadeh, L.A., "The concept of a linguistic variable and its application to approximate reasoning, Parts 1, 2, and 3," Information Sciences, 1975, 8:199-249, 8:301-357, 9:43-80.

[3] Ушаков К.З. Газовая динамика шахт. – М.: Горные науки, 2004.-481с.

[4] Абрамов Ф.А., Фельдман Л.П., Святный В.А. Моделирование динамических процессов рудничной аэрологии. Киев, Научная мысль, 1981. 380с.

[5] Леоненков А. В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. — СПб.: БХВ Петербург, 2005. — 736 с.: ил.

[6] Ковалев С.А., Азамат М.Н., Чепижко А.А. Моделирование объекта управления системы проветривания выемочного участка. Известия ТТИ ЮФУ – ДонНТУ. "Практика и перспективы развития партнерства в сфере высшей школы". В 2 кн. – Таганрог ТТИ ЮФУ.Кн.2. 2012, № 12. – с.8-12.

[7] Mamdani, E.H. and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," International Journal of Man-Machine Studies, Vol. 7, No. 1, pp. 1-13, 1975.

Отримано 21.06.2017

УДК 681.3

МАТЕМАТИЧНА МОДЕЛЬ ПОГЛИНАННЯ ВИПРОМІНЮВАННЯ ДЛЯ ОПТОЕЛЕКТРОННОГО ТРАНСПАРАНТА З ДВОХВИЛЬОВИМ КЕРУВАННЯМ

Г. Л. Лисенко, І. В. Кошельна, О. В. Бурмакіна
Вінницький національний технічний університет

Розглядається математична модель поглинання світлового потоку при проходженні через електроабсорбційні модулятори на основі напівпровідникової структури GaAs з квантовими ямами AlGaAs в залежності від довжини світла та прикладеної напруги.

Поглинання випромінювання напівпровідником транспарантом може змінюватися в залежності від різних факторів, основними з яких є довжина хвилі керуючого випромінювання та величина прикладеної напруги. Ці залежності є основою для здійснення керування транспарантом на напівпровідникових структурах з квантовими ямами (КЯ). Тому отримаємо аналітичний вигляд математичної моделі залежності поглинання випромінювання транспарантом від керуючої довжини хвилі та прикладеної напруги.

Оскільки елементною базою для транспаранта було обрано електроабсорбційні модулятори OCEAM [1], а особливістю їх побудови є те, що вони складається з двох вертикально вирощених один на одному p-i-n-діодів. Верхній p-i-n-діод є фотоприймачем (ФП) фотодіодом для керуючого випромінювання, а нижній - модулятором для інформаційного променя EAM. Тому розглянемо окремо математичні моделі для фотодіода та модулятора (EAM).

З експериментальних даних [1, 2] відомо, що для електроабсорбційних модуляторів на напівпровідникових структурах з КЯ, які використовуються для розробки транспарантів, існують графічні характеристики, що описують залежності коефіцієнта поглинання випромінювання від довжини хвилі та прикладеної напруги. Дані експериментальні результати отримані шляхом вимірювання поглинання випромінювання [3] напівпровідникової структури GaAs з квантовими ямами AlGaAs. Результати цих експериментів наведені на рис.1.

Для ФП керуючого випромінювання, який є p-i-n-діодом, справедливі залежності для струмів, що породжуються електронами і дірками. При освітленні транспаранту на основі OCEAM зі сторони p-області струм, що породжується дірками, визначатиметься для $0 \le t \le \tau_1$, як

$$I_p(t) = \frac{q\upsilon_{h,satur}}{S}\left[1 - e^{-\alpha(\upsilon_{h,satur}(\tau_\rho - \tau_1))}\right], \quad (1)$$

та для $\tau_1 < t \le \tau_p$ як

$$I_p(t) = \frac{q\upsilon_{h,satur}}{S}\left[e^{-\alpha(\upsilon_{h,satur}(t-\tau_1))} - e^{-\alpha(\upsilon_{h,satur}(\tau_p - \tau_1))}\right], \quad (2)$$

де $\tau_p = (x_1 + x_2)/\upsilon_{h,satur}$;

$\tau_1 = x_1/\upsilon_{h,satur}$

S – товщина верхнього p-i-n-діода, яка складається з x_1, x_2, x_3;

$\upsilon_{h,satur}$- швидкість сатурації дірок;

x_1 – товщина шару, що межує з p-областю;

x_2 – товщина поглинаючого шару, що лежить після шару x_1.

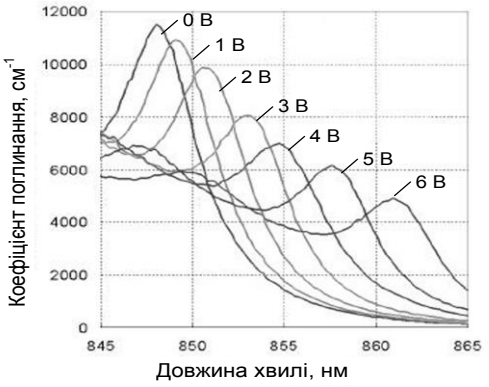


Рисунок 1 - Залежності коефіцієнта поглинання GaAs/AlGaAs при різних прикладених напругах [3]

Аналогічно до (1) і (2), при освітленні транспаранту зі сторони n-області струм для фотоприймача, що породжується електронами визначатиметься для $0 \le t \le \tau_2$, як

$$I_n(t) = \frac{q\upsilon_{e,satur}}{S}\left[1 - e^{-\alpha x_2}\right] \quad (3)$$

та для $\tau_2 < t \le \tau_n$ як

$$I_n(t) = \frac{q\upsilon_{e,satur}}{S}\left[1 - e^{-\alpha(L - \upsilon_{e,satur}t - x_1)}\right], \quad (4)$$

де $\tau_2 = x_3/\upsilon_{e,satur}$;

$\tau_n = (x_2 + x_3)/\upsilon_{e,satur}$

$\upsilon_{e,satur}$- швидкість сатурації електронів;

x_3 – товщина непоглинаючого шару, що лежить після шару x_2.

Цей час відгуку імпульсу може бути керований будь-яким потоком даних для генерації струму фотодіодного модулятора.

Чутливість ФП описується формулою (5) та залежить від струму як функції від довжини хвилі випромінювання, який протікає через нього [4]

$$S(\lambda^{ctrl}) = \frac{I_{ph}(\lambda^{ctrl})}{P_{in}^{\lambda^{ctrl}}}, \tag{5}$$

де $I_{ph}(\lambda^{ctrl})$ - струм, що протікає через верхній p-i-n-діод;

$P_{in}^{\lambda^{ctrl}}$ – потужність керуючого випромінювання.

Для виведення математичної моделі поглинання інформаційного випромінювання модулятором (нижнім p-i-n-діодом) необхідно застосувати залежність чутливості ЕАМ від поглинання цього випромінювання:

$$S(\lambda,V) = \frac{e\cdot\eta\ \cdot\lambda}{h\cdot c}\left[1-e^{-\alpha(\lambda,V)L}\right]. \tag{6}$$

Крім того, можна переписати (5) для ЕАМ, щоб оцінити залежність чутливості від струму, що протікає через ЕАМ:

$$S(\lambda,V) = \frac{I_{EAM}(\lambda,V)}{P_{in}^{\lambda}}. \tag{7}$$

Співставивши формули (6) і (7) можливо виявити залежність поглинання інформаційного випромінювання від прикладеної напруги:

$$\frac{I_{EAM}(\lambda,V)}{P_{in}^{\lambda}} = \frac{e\cdot\eta\ \cdot\lambda}{h\cdot c}\left[1-e^{-\alpha(\lambda,V)L}\right]. \tag{8}$$

З (8) не важко знайти, що

$$e^{-\alpha(\lambda,V)L} = 1-\frac{I_{EAM}(\lambda,V)\cdot h\cdot c}{P_{in}^{\lambda}\cdot e\cdot\eta\ \cdot\lambda}. \tag{9}$$

Застосувавши операцію логарифмування до (9), отримаємо модель залежності поглинання від прикладеної напруги та довжини хвилі випромінювання α (λ,V)

$$\alpha(\lambda\ ,V) = -\frac{1}{L}\ln\left(1-\frac{I_{EAM}(\lambda,V)hc}{e\eta\lambda\ P_{in}^{\lambda}}\right), \tag{10}$$

де η – квантова ефективність;

L – довжина модулятора;

P_{in}^{λ} – потужність інформаційного випромінювання;

h – стала Планка;

c – швидкість світла;

e – заряд електрона;

$I_{EAM}(\lambda,V)$ – залежність струму, що протікає через електроабсорбційний модулятор від прикладеної до нього напруги та довжини хвилі інформаційного випромінювання.

Із (10) випливає, що поглинання, яке є функцією довжини хвилі і прикладеної напруги, залежить також і від струму, що проходить через модулятор.

Для перевірки відповідності знайденого аналітичного вигляду математичної моделі (10) експериментальним даним [1, 2] необхідно знати залежність фотоструму, що протікає через електроабсорбційний модулятор, від довжини хвилі та прикладеної напруги.

Залежності фотоструму від довжини хвилі керування та напруги було експериментально досліджено Д.Міллером в роботі [5]. Використовуючи комп'ютерні методи інтерполяції таблично заданих значень та математичні формули полінома Лагранжа, було виведено аналітичні залежності фотоструму від довжини хвилі та прикладеної напруги.

Результат залежності $I_{EAM}(\lambda_i,V)$ має вигляд

$$I_{EAM}(\lambda,V) = 2\cdot10^{-9}\cdot\lambda\cdot10^{6} - 7\cdot10^{-8}\cdot\lambda\cdot10^{5} + 10^{-6}\cdot\lambda\cdot10^{4} - 2\cdot10^{-5}\cdot\lambda\cdot10^{3} + \\ + 9\cdot10^{-3}\cdot\lambda\cdot10^{2} - 0{,}3\cdot\lambda - 8\cdot10^{-4}V + 0{,}4036. \qquad (11)$$

Підставляючи аналітичну залежність фотоструму від довжини хвилі та напруги (2.20) у (2.19), отримаємо остаточний вигляд математичної моделі поглинання випромінювання від довжини хвилі і прикладеної напруги:

$$\alpha(\lambda,V) = -\frac{1}{L}\ln\left(1 - \frac{hc}{e\eta\lambda\ P_{inc}}\cdot\left[2\cdot10^{-9}\cdot\lambda\cdot10^{6} - 7\cdot10^{-8}\cdot\lambda\cdot10^{5} + 10^{-6}\cdot\lambda\cdot10^{4} - \right.\right. \\ \left.\left. - 2\cdot10^{-5}\cdot\lambda\cdot10^{3} + 9\cdot10^{-3}\cdot\lambda\cdot10^{2} - 0{,}3\cdot\lambda - 8\cdot10^{-4}V + 0{,}4036\right]\right) \qquad (12)$$

Дослідимо математичну модель (12) на зміну довжини хвилі і прикладеної напруги. Зокрема, проаналізуємо, як зміниться поглинання ЕАМ при $\lambda\to0$ та $\lambda\to\infty$. Проте такі записи для λ необхідно уточнити. Оскільки робочою областю довжин хвиль на напівпровідниках є ближній ІЧ діапазон, то обмежимося його межами, оскільки поглинання на інших довжинах хвиль для напівпровідників

не відбуватиметься. Крім того, з експериментальних даних відомо, що не лінійність залежності поглинання від довжини хвилі існує в діапазоні до 0,86 мкм. Тому оберемо це значення як верхню межу.

Вважатимемо, що при $\lambda\to 0$ маємо $\lambda\to 0{,}73$ мкм, а при $\lambda\to\infty$ маємо $\lambda\to 0{,}86$ мкм. Дослідимо чому будуть рівні границі $\lim\limits_{\lambda\to 0}\alpha(\lambda,V)$ та $\lim\limits_{\lambda\to\infty}\alpha(\lambda,V)$ при потужності інформаційного сигналу $N\cdot 10^{-3}$ Вт. Отже,

$$\lim_{\substack{\lambda\to 0\\ V\to const}}\alpha(\lambda,V)=\lim_{\substack{\lambda\to 0.73\\ V\to 1B}}\alpha(\lambda,V)=-\frac{1}{3\cdot 10^{-4}}\ln\left(1-\frac{0.8489hc}{0.73\cdot 10^{-3}e\eta P_{in}^{\lambda}}\right)=N\cdot 10^{2}$$

.

З отриманих результатів видно, що для довжини хвилі 0,73 мкм величина поглинутого випромінювання буде недостатньою для здійснення модуляції інформаційного пучка.

Аналогічно, для $\lambda\to\infty$ справедливо

$$\lim_{\substack{\lambda\to\infty\\ V\to const}}\alpha(\lambda,V)=\lim_{\substack{\lambda\to 0.86\\ V\to 1B}}\alpha(\lambda,V)=-\frac{1}{3\cdot 10^{-4}}\ln\left(1-\frac{0.8912hc}{0.86\cdot 10^{-3}e\eta P_{in}^{\lambda}}\right)=N\cdot 10^{2}$$

.

Тобто, і для довжини хвилі 0,86 мкм також величина поглинутого випромінювання буде недостатньою для здійснення модуляції інформаційного пучка.

Дослідимо тепер частинний випадок всередині обраного діапазону 0,73....0,86 мкм. Обчислимо $\lim\limits_{\lambda\to 848нм}\alpha(\lambda,V)$ при аналогічній вхідній потужності:

$$\lim_{\substack{\lambda\to 0.848\\ V\to 1B}}\alpha(\lambda,V)=-\frac{1}{3\cdot 10^{-4}}\ln\left(1-\frac{1.662hc}{0.848\cdot 10^{-2}e\eta P_{in}^{\lambda}}\right)=N\cdot 10^{4}.$$

Отже, видно, що для довжини хвилі 848 нм величина поглинутого випромінювання перевищує попередні результати на 2 порядки і буде достатньою для здійснення модуляції інформаційного пучка. Тому доцільно досліджувати значення поглинутого випромінювання саме в цьому діапазоні.

Аналогічно дослідимо математичну модель (12) при $V\to 0$ та $V\to\infty$. Випадок $V\to\infty$ обмежимо до $V\to 10$ В. Оскільки напруженість поля напівпровідникових лазерів, що застосовуються в якості джерел випромінювання для ЕАМ, становить 10^4....10^5 В/см. Перерахувавши значення напруженості на товщину ЕАМ (яка досягає сотень мікрон), отримаємо $V\to 10$ В.

Отже, при $V\to 0$ для аналогічних значень потужності випромінювання маємо

$$\lim_{\substack{\lambda\to 0.848\\ V\to 0}} \alpha(\lambda,V) = -\frac{1}{3\cdot 10^{-4}} \ln\left(1-\frac{1.582hc}{0.848\cdot 10^{-2} e\eta P_{in}^{\lambda}}\right) = N\cdot 10^{4}.$$

При V$\to$10 отримуємо

$$\lim_{\substack{\lambda\to 0.848\\ V\to 10}} \alpha(\lambda,V) = -\frac{1}{3\cdot 10^{-4}} \ln\left(1-\frac{0.0862hc}{0.848\cdot 10^{-2} e\eta P_{in}^{\lambda}}\right) = N\cdot 10^{2}.$$

З останнього виразу видно, що зі збільшенням прикладеної до модулятора напруги зменшується величина поглинутого випромінювання.

Дослідження математичної моделі (12) показало, що вона неоднаково справедлива для різних значень λ і V. Тому необхідно ввести обмеження для її застосування, які випливають з проведеного аналізу:

1) застосування математичної моделі можливе при низьких напругах (від 0 до 10 В);

2) адекватність результатів цієї моделі можлива в діапазоні довжин хвиль 0,73 – 0,86 мкм, що випливає з аналізу експериментальних даних [5, 6].

ЛІТЕРАТУРА

1. Vijit Sabnis. Optically-controlled electroabsorption modulators for future generation optical networks// A dissertation submitted to the department of applied physics and the committee on graduate studies of Stratford University in partial fulfillment of the requirements for the degree of Doctor of Philosophy, 2003. – 166 p.

2. Noah Charles Helman. Optoelectronic modulators for optical interconnects// A dissertation submitted to the department of applied physics and the committee on graduate studies of Stratford University in partial fulfillment of the requirements for the degree of Doctor of Philosophy, 2005. – 178 p.

3. Bianca Elizabeth Nelson Keller. Wavelength division multiplexed optical interconnects using short pulses// A dissertation submitted to the department of applied physics and the committee on graduate studies of Stanford University in partial fulfillment of the requirements for the degree of Doctor of Philosophy, 2002. - P.48.

4. Gordon Arthur Keller. Optical interconnects to silicon CMOS: integrated optoelectronic modulators and short pulse systems// A dissertation submitted to the department of applied physics and the committee on graduate studies of Stanford University in partial fulfillment of the requirements for the degree of Doctor of Philosophy, 2002. – 182 p.

5. Excitons in resonant coupling of quantum wells/ A.M.Fox, D.A.B.Miller, G.Livescu [and other]// Physical Review B. – 1990/ - Vol.42, №3. – P. 1841-1844.

6. Оптоелектронні напівпровідникові транспаранти для швидкодійних спеціалізованих обчислювачів : монографія / Г. Л. Лисенко, І. В. Мялківська. – Вінниця : ВНТУ, 2015. – 140 с.

Отримано 26.06.2017

УДК 004.67:004.8:615.12

ВИКОРИСТАННЯ МЕТОДІВ І АЛГОРИТМІВ ОБРОБКИ ВЕЛИКИХ ОБСЯГІВ ДАНИХ ПРИ РІШЕННІ ЗАДАЧ ФАРМАЦЕВТИЧНОЇ ЛОГІСТИКИ

Н.О.Маслова, О.Л.Половинка

Донецький національний технічний університет,

Україна, Покровськ

У статті доведена актуальність та необхідність пошуку ефективних алгоритмів, у тому числі паралельних, в застосуванні до фармацевтичних даних великого обсягу. Наведено приклади існуючих пакетів обробки, проаналізовано їх основні можливості та недоліки. Виділено напрямки для подальших досліджень.

Ключові слова: Data Mining, інтелектуальний аналіз даних, фармацевтична логістика, паралельні алгоритми, розподілені системи

У світі сучасних технологій особливо гостро стоїть питання збору та обробки даних великих обсягів. Одним з ефективних способів роботи з даними є технології інтелектуального аналізу даних, або Data Mining. Це потужний апарат сучасної бізнес-аналітики та дослідження даних для виявлення прихованих закономірностей і побудови моделей [1].

Використання алгоритмів Data Mining є перспективним і результативним у багатьох сферах, і, зокрема, в галузях медицини і фармакології. Так, наприклад, вчені з Стенфордського університету (Каліфорнія), проаналізували базу даних FDA (Food and Drug Administration - Управління та контроль за продуктами і лікарськими препаратами) засобами Data Mining та змогли виявити 47 раніше невідомих несприятливих взаємодій лікарських препаратів [1].

Тож, результати, отримані при аналізі значних обсягів даних мають особливу цінність. У зв'язку з цим можна виділити дві основні **проблеми аналізу**:

- розподіленість: збір інформації з різних джерел та необхідність зберігання її в розподіленому сховищі;
- продуктивність: аналіз обсягів інформації, що вимірюються терабайтами, може займати неприйнятний

для аналізу час, й вимагати великих обчислювальних потужностей.

В [1] стверджується, що обидві проблеми можуть вирішуватися за рахунок паралельних (Parallel Data Mining) і (або) розподілених (Distributed Data Mining) алгоритмів Data Mining.

Й дійсно, на поточному етапі основні зусилля дослідників спрямовані на розпаралелювання окремих алгоритмів аналізу і їх подальшу оптимізацію. Однак, у зв'язку з тим, що алгоритми орієнтовані на певне середовище обчислень, при перенесенні знайдених рішень в інші умови, вони стають неефективними.

Сказане вище дозволяє зробити висновок, що дослідження в області оптимізації алгоритмів Data Mining, створення версій для роботи в паралельних і розподілених середовищах є **актуальною**, сучасною і практично необхідною задачею.

Метою роботи є пошук методів і алгоритмів ефективної обробки великих обсягів даних при вирішенні завдань фармацевтичної логістики з урахуванням можливості паралельної і розподіленої обробки даних.

Основна частина. При роботі з великими обсягами даних (Big Data Environment) традиційно виділяють задачі збору, зберігання і обробки. При вирішенні задач обробки використовуються методи оперативного (OLAP) і інтелектуального (Data Mining) аналізів. Крім того, для певних типів даних виділяють методи візуального і текстового аналізу (Visual Mining, Text Mining). У даній роботі передбачається використання засобів Data Mining.

Основними видами алгоритмів інтелектуального аналізу даних є алгоритми класифікації, кластеризації та побудови асоціативних правил. Саме класичні алгоритми і їх модифікації служать основою для побудови у подальшому паралельних та розподілених структур.

Найбільш часто використовувані алгоритми Data Mining і їх короткі характеристики наведені в таблиці 1. Безумовно, перелік алгоритмів може бути подовжено й розширено, але вказані типи – алгоритми класифікації, кластеризації та побудови асоціативних правил є базовими.

Загалом, в області оптимізації та розпаралелювання алгоритмів інтелектуального аналізу існує значна кількість розробок як методичного, так і алгоритмічного напрямку. Спочатку в сукупність підходів та інструментів включалися засоби масово-паралельної обробки невизначено структурованих даних (NoSQL).

Таблиця 1. Аналіз поширених Data mining-алгоритмів

	Призначення	**Переваги/недоліки**
C4.5	Будує класифікатор в формі дерева рішень (створює дерево рішень, яке використовується для прийняття рішень) Реалізовано в: методах data mining. Orange (Java)	Переваги: - проста інтерпретація; - висока швидкість роботи. Особливість: можливість обробки як дискретних, так і безперервних даних
SVM (Support vector machine)	**Метод опорних векторів.** Будує гіперплоскість, щоб класифікувати дані по 2 класам, але не використовує дерева рішень. Реалізовано в: scikit-learn, MATLAB, libsvm	Переваги: - простота інтерпретації; - висока швидкість роботи. Недоліки: - необхідність вибору ядра і досить часто - погана якість інтерпретації
Метод к-средних	Створює к-груп з набору об'єктів таким чином, щоб члени групи були найбільш однорідними (кластерний аналіз) Реалізовано в: Apache Mahout; Julia; R; SciPy; Weka; MATLAB; SAS	Переваги: ефективність; можливість використання для попереднього розбиття на групи великого набору даних. Недоліки: чутливість до «викидів» та початкового вибору центрів тяжіння.
Алгоритм Apriori	Шукає асоціативні правила і застосовується по відношенню до баз даних, що містять величезну кількість транзакцій Реалізовано в: ARtool, Weka и Orange	Простий, зрозумілий, легко реалізується, має безліч модифікацій. Недоліки: ресурсомісткість; високі витрати часу на виконання
PageRank	Алгоритм ранжування за посиланням. Реалізовано в: C++ OpenSource PageRank; Python PageRank; пакет igraph в R	Перевага - надійність. Недолік - складність отримання релевантного вхідного посилання
AdaBoost	алгоритм посилення класифікаторів Реалізовано в: scikit-learn ; ICSIBoost; gbrm (Generalized Boosted Regression Models)	Переваги: гнучкість; легка пристосованість; можливість роботи з будь-якими алгоритмами, що навчаються.

Пізніше вони були розширені алгоритмами серії MapReduce разом з програмними каркасами, що їх реалізують та бібліотекми проекту Hadoop. Надалі до інструментів обробки великих даних стали відносити різноманітні інформаційно-технологічні рішення, що забезпечують можливості з обробки надвеликих масивів даних [3].

З числа базових можна назвати роботу [4] з описом бібліотеки алгоритмів інтелектуального аналізу даних для виконання в паралельних і розподілених середовищах DXelopes. А в [5] - описано її комерційний варіант, орієнтований на виконання алгоритмів в паралельних і розподілених середовищах і використання хмарних технологій. За заявою розробників реалізовано каркас бібліотеки, базові алгоритми і адаптери для виконання в багато потоковому і розподіленому середовищі. Автори пропонують методику розпаралелювання алгоритмів інтелектуального аналізу даних на основі блокової структури. Дана методика враховує характеристики середовища виконання паралельних алгоритмів, типи розподілу аналізованих даних, а також особливості алгоритму, який передбачено розпаралелити.

Одним з нових перспективних напрямів в розподілених системах, є Grid computing - обробка інформації в Супер-мережі (Грід). Існує спільний проект українських та американських вчених «Розробка ефективних GRID-технологій екологічного моніторингу на основі супутникових даних», під керівництвом Куссуль Н.М., в партнерстві з Дж. Марковським з Університету штату Мен [6].

Дослідження і результати.

В пропонованій роботі в якості вхідних передбачається використання даних аптечних мереж з метою отримання аналітичних залежностей, необхідних для вирішення завдань фармацевтичної логістики. Вони характеризуються великими обсягами інформації, розподіленою структурою, наявністю спеціалізованих пакетів збору і обробки даних початкового рівня.

У матеріалах [2] наведено опис особливостей фармацевтичної логістики, перераховані двадцять основних принципів, що її характеризують і підтверджують складність і багатовимірність завдань обробки фармацевтичних даних. Ці дані можуть бути віднесені до Big Data - до категорії обробки структурованих і неструктурованих даних великих обсягів.

Важливість і актуальність аналізу великих обсягів даних настільки значна, що в різних програмних пакетах обробки

промислового призначення були зроблені спроби максимально врахувати потреби фармацевтичних підприємств і розробити аналітичні форми. Проведемо аналіз в деяких з них.

Програмний Комплекс «Аптека-Цель».

Особливістю програмного комплексу є те, що він розроблявся спільно з ключовими користувачами системи, тобто при розробці були максимально враховані і задоволені потреби співробітників аптек, має максимально простий інтерфейс, що дозволяє в найкоротші терміни обслуговувати клієнтів. В роботі комплексу використовується реляційна база даних типу dbase. У зв'язку з цим можна відразу виявити два мінуси системи: незахищеність баз даних і тривалий час обробки великих масивів даних. Однак, при цьому система може працювати з мінімальними потужностями і в цілому є досить ефективною. Що стосується аналізу, то він представлений величезною кількістю різних друкованих форм, які створюються за запитами ключових користувачів методом «локальна вибірка». Плюсом такого способу є можливість отримати абсолютно будь-яку цікаву для користувача інформацію. Мінусом є неможливість одночасної обробки розподілених даних, невикористання паралельних методів, складність роботи з угрупованнями і фільтрами даних. У таблиці 2 наведені часові витрати на формування двох відомостей засобами ПК «Аптека-Цель». Особливістю даних відомостей є те, що крім вибірки даних при побудові відбуваються різні розрахунки за підрахунком поточного залишку товарів, а також, розрахунок ряду характеристик, необхідних для коректного аналізу роботи аптечної мережі.

Таблиця 2. Час побудови аналітик з пакетом «Аптека-Цель»

Назва аналітики	Обсяг (к-ть записів)	Час побудови
Відомість залишків по мережі	56 921	30-60 хвилин
Моніторинг цін на товари першої необхідності	4 267	До 10 хвилин

Програмний комплекс «Моріон-Аптека».

Даний програмний продукт з'явився порівняно недавно і зараз знаходиться в стадії активної розробки. Він має стандартний багатовіконний табличний інтерфейс і величезну кількість параметрів, які теоретично дозволяють користувачеві максимально зручно налаштувати інтерфейс програми. Однак, на практиці цей метод себе не виправдовує, оскільки, як правило, кваліфікація співробітників і

завідувачів аптекою недостатньо висока для повноцінного використання даних можливостей. Програма використовує кросплатформну систему управління базами даних Firebird. Це дозволяє забезпечити дані ефективним захистом, однак, при цьому швидкість роботи системи значно знижується. Також, використання даної СУБД вимагає досить серйозних комп'ютерних потужностей. Аналіз в даній програмі можна умовно розділити на дві частини:

- стандартна аналітика, спочатку закладена в програмі, включає в себе як аналіз продажів, рух документів, залишків так і різні моніторинги, і побудову ТОП-списків;
- OLAP-аналіз, який включає в себе куб реалізації, куб залишків і куб документів.

Дані, які аналізуються, виводяться в інтерактивних таблицях, що є безсумнівним плюсом, оскільки це дозволяє користуватися фільтрами, угрупованнями, а також прибирати з екрану або виводити ті дані, аналіз яких необхідний у даний момент. OLAP-аналіз, в свою чергу, дозволяє побудувати різні призначені для користувача звіти, а також діаграми та графіки по ним. Однак, надані дані жорстко обмежені зрізами кубів і, при всьому своєму різноманітті, не дають повної картини і не дозволяють прийняти ряд важливих рішень. Також, не вирішена проблема часу обробки великих масивів даних. Час побудови аналітик за тривалий період з великою кількістю структур наведено в таблиці 3. Слід вважати, що в аналітиці не враховується ряд найважливіших розрахункових показників, наприклад, таких як націнка, тобто безпосередній дохід від продажів.

Таблиця 3. Час побудови аналітик в пакеті «Моріон-Аптека»

Назва аналітики	Обсяг (к-ть записів)	Час побудови
Відомість залишків по мережі	56 921	20-30 хвилин
Моніторинг цін на товари першої необхідності	4 267	До 5 хвилин

Ще одним програмним продуктом, що використовується досить широко, є автоматизована система управління підприємством Axapta. У багатьох мережах вона застостовується для закупівлі і відстеження руху товару. Однак, аналітичний апарат даної системи розвинений досить слабо, в зв'язку з чим вимагає використання додаткових засобів аналізу.

Висновки. Таким чином, аналіз деяких розповсюджених програмних продуктів для фармакологічних підприємств, дозволяє зробити висновок, що наведені системи не формують бажану для обробки розподілену структуру інформації та не включають методичні або алгоритмічні розробки, що необхідні для прийняття важливих рішень з логістики. Також, для кожної системи гостро стоїть питання швидкодії та ефективності обробки та аналізу великих обсягів даних. У цих умовах розробка і впровадження паралельних і розподілених алгоритмів Data Mining є необхідним й перспективним методом вирішення проблем аналізу фармацевтичних даних.

БІБЛІОГРАФІЧНІ ПОСИЛАННЯ

[1] Введение в современный Data Mining [електронний ресурс] http://statistica.ru/local-portals/data-mining/

[2] Половинка О.Л. Особенности управления логистическими потоками в фармацевтической сети / Половинка О.Л., Гизатулин А.М. Матеріали шістнадцятої всеукраїнської науково-практичної конференції "Інноваційний потенціал української науки - XXI сторіччя" (09 - 14 квітня 2012 року) [електронне видання] http://nauka.zinet.info/16/polovinka.php

[3] Jeffrey Cohen, Brian Dolan, Mark Dunlap, Joseph M. Hellerstein, Caleb Welton. MAD Skills: New Analysis Practices for Big Data. Proceedings of the VLDB'09 Conference, Lyon, France, August 2009.- p.24-28.

[4] Барсегян, А.А. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP / А.А.Барсегян, М.С.Куприянов, В.В.Степаненко, И.И.Холод. 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2007. – 384с.

[5] Big Data Environment. Библиотека алгоритмов интеллектуального анализа данных для выполнения в параллельных и распределенных средах DXelopes [електронний ресурс] http://www.eltech.ru/assets/files/nauka/razrabotki/Informatika-Big-Data.pdf

[6] Отдел космических информационных технологий и систем. Разработка эффективных GRID-технологий экологического мониторинга на основе спутниковых данных [електронне видання] http://inform.ikd.kiev.ua/index.php?path=/ru/projects/UNTC

Отримано 27.06.2017

УДК 004.382

МУЛЬТИПЛАТФОРМЕННЫЙ РЕДАКТОР ДВУМЕРНЫХ ГРАФОВ ОБЩЕГО НАЗНАЧЕНИЯ В КОНТЕКСТЕ МОДЕЛЬНОЙ ПРЕДМЕТНОЙ ОБЛАСТИ ГРАФОВ ШАХТНЫХ ВЕНТИЛЯЦИОННЫХ СЕТЕЙ

А.Л.Масюк, асп.; А.Н.Мирошкин, к.т.н.
ДонНТУ, г. Покровск

Розглядається мультиплатформний редактор графів загального призначення як універсальний діалоговий засіб підтримки першого етапу моделювання шахтного мережевого об'єкту. Орієнтований на візуальне редагування топології і параметризацію графів, редактор може знайти застосування в контексті практично будь-якої модельної предметної області, основою яких є двовимірні графи довільній зв'язності.

Большая размерность сетей шахтных вентиляционных сетей (число ветвей более 100, число узлов более 50), многосвязность физических контролируемых и регулируемых параметров, иерархичность расположения регулирующих органов, нелинейность аэродинамических характеристик вентиляторов и выработок, распределенность параметров выработок и выработанных пространств, сильная взаимосвязь между аэро- и газодинамическими переходными процессами, наличие стохастических и детерминированных возмущений являются основными показателями сложности ШВС [1-3]. Поэтому актуальной является задача компьютерной поддержки формирования параллельных моделей ШВС и разработки соответствующих диалоговых алгоритмов, которые бы позволили эксперту предметной области наиболее интуитивным образом создавать данные модели и эффективно работать с ними.

В статье рассматривается мультиплатформенное приложение-редактор графов общего назначения, как универсальное диалоговое средство поддержки первого этапа моделирования шахтного сетевого объекта [2]. Базовый функциональный набор включает в себя:

- редактор топологии графа абстрактной МПО;
- систему управления атрибутами модели данных;
- систему персистентного хранения моделей данных;
- модули импорта моделей данных из других форматов;
- интерфейсы с внешними модулями (топологический анализ, симулятор и др.);

- графический пользовательский интерфейс (рис.1).

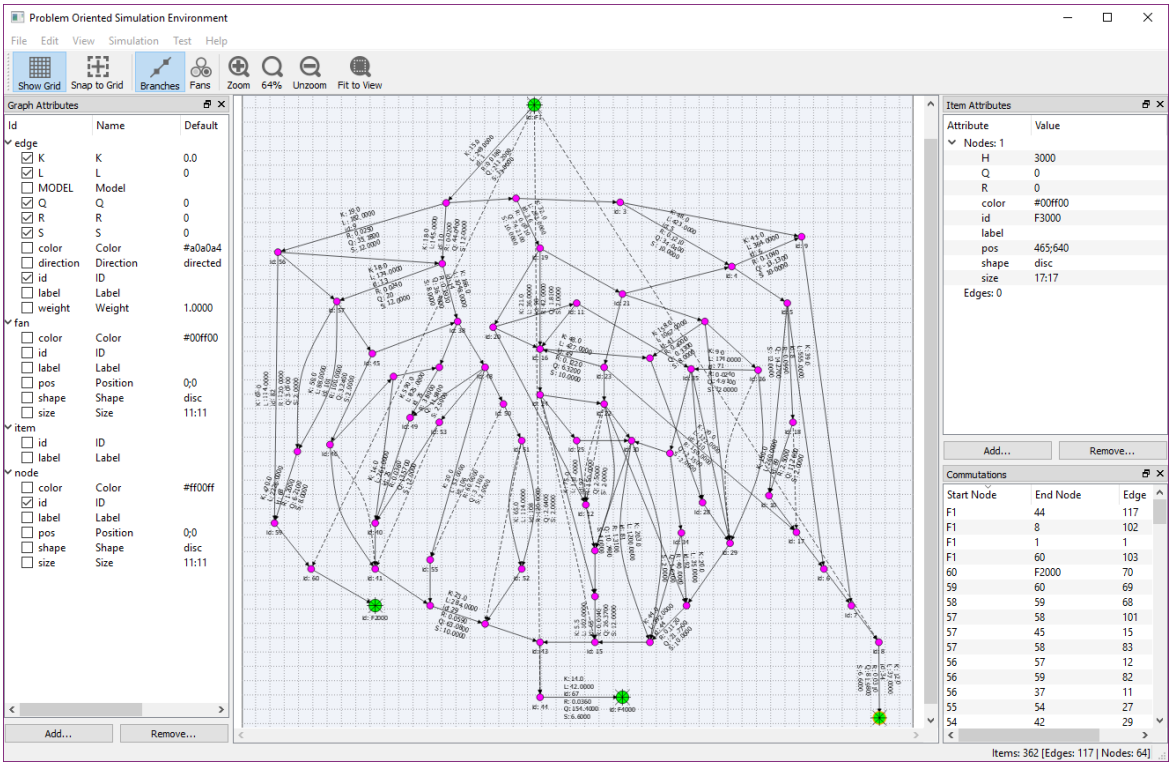


Рисунок 1 - Интерфейс приложения с загруженной моделью графа ШВС (117 ветвей, 3 вентилятора), ОС Windows 10

Абстрактная МПО топологии графа представлена двумя основными типами объектов: узлы (node) и ветви (edge). Каждый объект обладает свойствами (флагами), позволяющими конфигурировать его базовое поведение (видимость, возможность перемещения, возможность выбора, соединение с объектами других типов и т.д.), уникальным идентификатором, а также атрибутами (как предопределёнными, так и задаваемыми пользователем). Поведение базовых объектов может быть расширено либо программным способом (наследование), либо с использованием системы управления атрибутами (как программно, так и в определённой мере непосредственно через пользовательский интерфейс приложения).

Множество узлов и соединений между ними образуют *подмодель топологии графа*. Важным свойством топологии графа является его ориентация; в настоящее время приложением поддерживаются три вида ориентации, которые могут быть комбинированы в пределах одной подмодели (для каждой ветви отдельно): ненаправленный граф

(undirected), направленный граф (directed) и граф, имеющий обоюдные направления ориентации некоторых ветвей (mutual directed).

Система управления атрибутами является основным модулем, отвечающим за пользовательскую параметризацию. Модель атрибутов представлена несколькими уровнями приоритетов:

- локальные атрибуты объектов (узлов и ветвей);
- атрибуты классов объектов (значения по умолчанию);
- глобальные атрибуты графа (сцены).

Локальные атрибуты имеют наивысший приоритет и могут быть установлены для конкретных объектов. Их областью действия является только тот объект, для которого они заданы. При копировании объекта, объект-копия по умолчанию перенимает все локальные атрибуты объекта-источника.

Атрибуты классов устанавливаются для конкретных классов объектов (как предопределённых − node, edge, так и созданных пользователем). Они обладают свойством наследования (аналогично объектам ЯП C++) и используются, как правило, для задания свойств по умолчанию для вновь создаваемых объектов. Более низкий приоритет, чем у локальных атрибутов, означает, что изменение атрибута класса не приведёт к изменению аналогичного локального атрибута объекта (т.е. локальные атрибуты «перекрывают» собой атрибуты класса). Отсутствие у объекта конкретного локального атрибута означает, что для объекта будет актуален аналогичный атрибут его класса (пример: атрибут «color=red» для класса node означает, что все узлы, не имеющие локального атрибута «color», будут отображаться красным цветом).

Глобальные атрибуты распространяют область своего действия на весь граф, но в то же время имеют самый низкий приоритет (пример: ориентация графа может иметь значение «undirected», однако конкретные ветви могут быть направленными: «directed», что определяется их локальными атрибутами).

На основе атрибутов реализуются *подмодель визуализации* и *подмодель пользовательских параметров*, а также часть подмодели топологии графа, отвечающая за направления ветвей.

Подмодель визуализации отвечает за представление модели графа в пользовательском интерфейсе и включает в себя такие атрибуты, как цвет объектов (color), форма узлов (shape), их размеры (size) и координаты в единицах сцены (position). Для ветвей, к примеру, атрибутами визуализации являются вес (weight) и стиль штриховки

(style). Базовые классы, реализующие абстрактную МПО графа (CNode, CConnection), обладают возможностью извлекать стандартные визуальные атрибуты из подмодели визуализации и отрисовывать объекты на экране, исходя из их значений (так, «вес» ветви по умолчанию влияет на толщину её изображения). Классы-наследники могут произвольным образом расширять набор поддерживаемых атрибутов визуализации, реализуя собственные методы отрисовки.

Подмодель пользовательских параметров по умолчанию не имеет содержания и определяется только в контексте конкретной МПО. Каждому объекту графа может быть присвоено неограниченное количество параметров, тип и значения которых задаются экспертом МПО для конкретных моделей. Как пример атрибута, который может быть определён пользователем для произвольной МПО, может служить дополнительное текстовое описание объекта (метка, комментарий). Редактор имеет встроенную поддержку для отображения и изменения такого текста. На рис. 2 представлена схема абстрактной МПО графа с разбиением её на подмодели топологии, визуализации и пользовательских параметров.

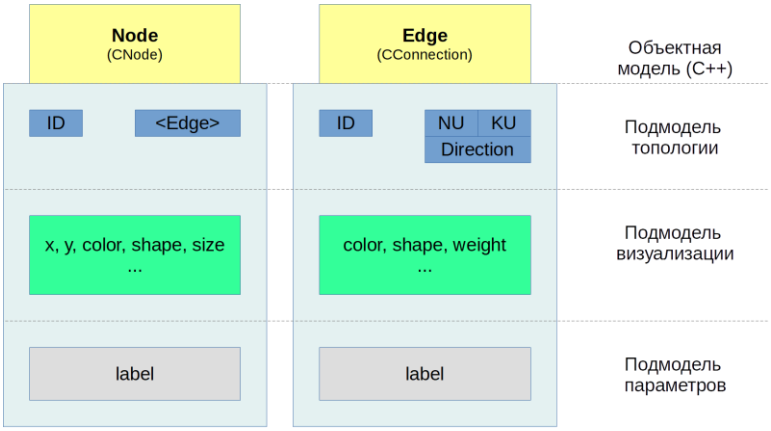


Рисунок 2 - Абстрактная МПО графа и подмодели, входящие в неё

Система персистентного (постоянного) *хранения моделей данных* подразумевает их выгрузку из приложения-редактора на внешний носитель в виде файла с возможностью последующей загрузки для продолжения работы. Концепция персистентного хранения моделей данных основана на идее универсального формата [4], которая предполагает единообразный доступ ко всем сохраняемым подмоделям.

Также в текущей версии приложения реализован импорт ограниченного множества тэгов из форматов GraphML [5] и GEXF [6].

Объектная архитектура визуального редактора графов основана на концепции т.н. «графической сцены» мультиплатформенного фреймворка Qt [7], обладающей базовой функциональностью для выделения и перемещения графических примитивов. Архитектура и API модуля редактирования графов были спроектированы таким образом, чтобы позволить максимально простыми (с точки зрения разработчкика) способами расширять абстрактную МПО, реализуя конкретные целевые МПО. Так, для использования приложения в контексте МПО графа ШВС потребовались следующие расширения абстрактной МПО:

- класс CBranchConnection, отнаследованный от CConnection, представляет ветвь графа ШВС. Он поддерживает визуальное отображение типа модели ветви (выработка, утечка, ветвь с РРВ/ГРРВ), задаваемой с помощью соответствующих атрибутов.

- класс CFanNode, отнаследованный от CNode, представляет физический объект-вентилятор. Как и CBranchConnection, CFanNode также реализует собственное визуальное отображение.

- добавлены атрибуты, задающие физические параметры объектов ШВС (длина ветви, депрессия, расход воздуха и т.д.). Типы атрибутов могут вариироваться в зависимости от входных данных, требуемых для решения конкретной задачи моделирования.

Следующим этапом [2] является передача созданной модели в модуль, реализующий процесс моделирования. В рамках исследуемой ПОМС было разработано несколько моделирующих алгоритмических модулей (симуляторов), каждый из которых ориентирован на решение определённой задачи. Основная задача ПД на данном этапе − предоставление пользователю универсального интерфейса, позволяющего единым образом задать основные параметры и метод моделирования (с учётом возможностей контретного симулятора и специфики решаемой задачи), после чего осуществить передачу созданной модели на моделирование. Это было достигнуто реализацией дополнительных интерфейсных модулей, принимающих на вход модель в универсальном формате и предоставляющих графический интерфейс пользователя для задания параметров моделирования. Полученные входные данные преобразовываются интерфейсным модулем в формат, требуемый конкретным симулятором, после чего запускается процесс моделирования, используя API симулятора. В ходе моделирования

осуществляется обратная связь с ПД (реализованная посредством паттерна Dependency Injection), что позволяет на каждом шаге моделирования получать и визуализировать промежуточные результаты.

Таким образом, в контексте работы над ПОМС было создано мультиплатформенное приложение-редактор графов абстрактной МПО с возможностью пользовательской параметризации объектов, реализованной с помощью развитой системы управления атрибутами. Далее абстрактная МПО была расширена до МПО графа ШВС, предоставляя эксперту средства для интуитивного создания и редактирования моделей. Реализовано сохранение модели графа ШВС и связанных с ней подмоделей в универсальном формате. Также реализован интерфейс между модулями симулятора и диалоговыми средствами ПОМС, что позволяет запускать на симуляцию параметризированную в редакторе модель графа ШВС, визуализировать промежуточные и финальные результаты моделирования.

ЛИТЕРАТУРА

1. **Святный В.А.** Проблеми паралельного моделювання складних динамічних систем. / Наукові праці Донецького національного технічного університету. Серія "Проблеми паралельного моделювання складних динамічних систем" – Донецьк: ДонНТУ. – 2011.

2. **Масюк А.Л.** Підсистема діалогу паралельного моделюючого середовища, орієнтованого на шахтні вентиляційні мережі. / Наукові праці Донецького національного технічного університету. Серія «Проблеми моделювання та автоматизації проектування» (МАП-2005). Випуск: 78 − Донецьк: ДонНТУ. − 2005. – 272 с.

3. **Масюк А.Л.** Диалоговые алгоритмы управления воздухораспределением в шахтной вентиляционной сети. / Вестник Донецкого Университета, Серия А. Естественные науки. №1. – Донецк, 2005. – с.395-403.

4. **Масюк А.Л.** Единое представление модели сетевого объекта для эффективной реализации алгоритмов диалоговой поддержки моделирования ШВС. / Наукові праці ДонНТУ. Випуск 93. Серія ІКОТ: - Донецьк: ДонНТУ, 2005. - с. 91-98.

5. **U.Brandes, M.Eiglsperger, I.Herman, M.Himsolt, M.S.Marshall**. GraphML Progress Report: Structural Layer Proposal. Proc. 9th Intl. Symp. Graph Drawing (GD '01), LNCS 2265, pp. 501-512. © Springer-Verlag, 2002.

6. GEXF 1.2draft Primer. (March 28, 2012). [Электронный ресурс]. – Режим доступа: https://gephi.org/gexf/1.2draft/gexf-12draft-primer.pdf

7. Qt Documentation: QGraphicsScene module. [Электронный ресурс]. – Режим доступа: http://doc.qt.io/qt-5/qgraphicsscene.html

Отримано 29.06.2017

УДК 004.932.4

ЕФЕКТИВНІСТЬ ОБЧИСЛЕНЬ ПРИ ПАРАЛЕЛЬНОМУ РОЗВ'ЯЗАННІ ЗАДАЧІ КОШІ НА ГРАФІЧНИХ ПРОЦЕСОРАХ ЕКСТРАПОЛЯЦІЙНИМИ МЕТОДАМИ

Л.П. Фельдман, І.А. Назарова, Я.А. Гризадубова, В.І. Костін
ДВНЗ «Донецький національний технічний університет»,
Україна

Запропоновано та досліджено ефективність паралельних методів чисельного розв'язання задачі Коші для систем звичайних диференційних рівнянь з контролем локальної похибки на основі методу локальної екстраполяції Річардсона з використанням графічних процесорів та технології CUDA

Key words: numerical methods, parallel modeling, local extrapolation of Richardson, graphic processors, CUDA technologies.

1. ВСТУП

Сучасні обчислювальні системи зазвичай містять області інтенсивної обробки даних з великою кількістю елементів з властивостями паралельності. Графічні процесори підвищують ефективність використання паралелізму, розподіляючи обчислення над сотнями ядр. Такий процес може забезпечити значне прискорення вирішення складних багатовимірних задач для реальних програмних додатків. Слід зауважити, що відсутність паралелізму вихідної задачі не тільки не дасть ніякого виграшу, але можливо навіть сповільнить обчислення, не зважаючи на паралелізм обчислювального пристрою.

Ядра CPU створені для виконання одного потоку послідовних інструкцій з максимальною продуктивністю, а GPU проектуються для швидкого виконання великої кількості паралельно виконуваних потоків інструкцій. Універсальні процесори оптимізовані для досягнення високої продуктивності єдиного потоку команд, при цьому доступ до пам'яті випадковий [1]. Обчислення на GPU вдають із себе тип гетерогенних обчислень, тобто, паралельні обчислення з декількома видами процесорних архітектур. В GPU обчисленнях багатоядерні і багатопотокові що дозволяє досягти більш високої продуктивності. У сучасних процесорів 4, 6, 8 або навіть 12 ядр, в той час як графічні процесори мають близько 512 ядр на одному пристрої.

На рисунку 1 зображені умовні архітектури графічного та центрального процесорів.

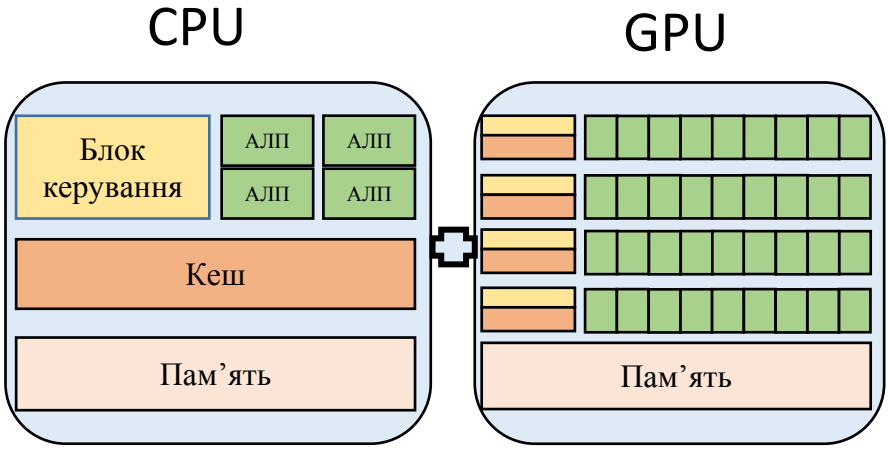


Рисунок 1 – Архітектура GPU і CPU

Є безліч відмінностей і в підтримці багатопоточності. CPU виконує 1-2 потоки обчислень на одному процесорному ядрі, а відеочіпи можуть підтримувати до 1024 потоків на кожному мультипроцесорі, яких в чіпі кілька штук. І якщо перемикання з одного потоку на інший для CPU коштує сотні тактів, то GPU переключає кілька потоків за один такт [1-3]. Крім того, центральні процесори використовують SIMD (одна інструкція виконується над численними даними) блоки для векторних обчислень, а відеочіпи застосовують SIMT (одна інструкція і кілька потоків) для скалярної обробки потоків. Для оптимальної роботи GPU потрібен спеціальний апаратно-програмний інтерфейс. Технологія CUDA – це програмно-апаратна обчислювальна архітектура NVIDIA, заснована на розширенні мови Сі, яка дає можливість організації доступу до набору інструкцій графічного прискорювача і управління його пам'яттю при організації паралельних обчислень.

Дана робота присвячена розробці та дослідженню ефективності паралельних методів розв'язання задачі Коші на основі технології локальної екстраполяції Річардсону (ЛЕР) [4-7]. Послідовна реалізація таких методів має великі обчислювальні витрати, тому з'являється сенс в паралельній реалізації методу ЛЕР, а також у підвищенні його ефективності за рахунок відображення на архітектуру з графічним пристроєм.

2. РЕАЛІЗАЦІЯ ТЕХНОЛОГІЇ ЛЕР З ВИКОРИСТАННЯМ ГРАФІЧНИХ ПРИСТРОЇВ

Динамічні задачі із зосередженими параметрами описуються системами звичайних диференційних рівнянь (СЗДР):

$$\bar{y}' = F(x, \bar{y}),\ \bar{y}(x_0) = \bar{y}_0, \tag{1}$$

де права частина у загальному випадку є нелінійна функція, що задає наступне відображення $F = \bar{f} : R \times R^m \to R^m$.

Екстраполяційна технологія для інтегрування СЗДР включає опорний чисельний метод розв'язання задачі Коші, послідовність сіток інтегрування, рекурентне правило обчислення значень наближеного розв`язку. Ефективність її застосування безпосередньо залежить від правильного вибору та поєднання всіх трьох складових. Для отримання сіток інтегрування застосовуються числові послідовності, утворені гармонійним рядом, рядом парних чисел, ступенями двійки [7-9]. Потенційно обчислення за технологією локальної екстраполяції містять три джерела внутрішнього паралелізму: системний паралелізм (обмежений розмірністю СЗДР); паралелізм екстраполяції (обмежений розміром таблиці екстраполяції); паралелізм опорного методу (мала ступінь паралелізму).

Для того щоб побудувати паралельний алгоритм локальної екстраполяції Річардсона (рисунок 2), в якості основної макрооперації вибирається одноразове обчислення деякої апроксимації рішення на основі k_0 -точкового опорного методу малого порядку. Кожна макрооперація є ітераційним процесом, виконаним максимальну можливу кількість раз, тобто L_0. Потім на базі формули Ейткена-Невілла формується інша частина екстраполяційної таблиці [4-5].

Алгоритм для графічного процесору є подібним до SIMD-моделі програмування, але є деякі відмінності. Нитки, які входять до складу блоків, фізично розподіляються на групи і виконуються паралельно, ці групи зазвичай можуть мати максимальну кількість ниток, встановлених виробниками графічного процесору. Нитки спільно використовують shared-пам’ять і потребують явної синхронізації. Блоки спільно використовують global-пам’ять. Графічні процесори мають додаткові механізми точнішого керування global- та shared-пам’яттю. Швидкість алгоритму суттєво залежить від того, на які саме групи будуть розподілені потоки та блоки (довжина блоку може бути 32/64/128 байт) та які типи даних будуть обрані (на кожен елемент має бути 2*N/16 звернень до глобальної пам'яті, та до 16 звернень одночасно до shared-пам’яті). Кожному блоку зіставляється частина

масиву даних. Блок копіює дані в shared-пам'ять. Проводить обчислення паралельно та зберігає результат.

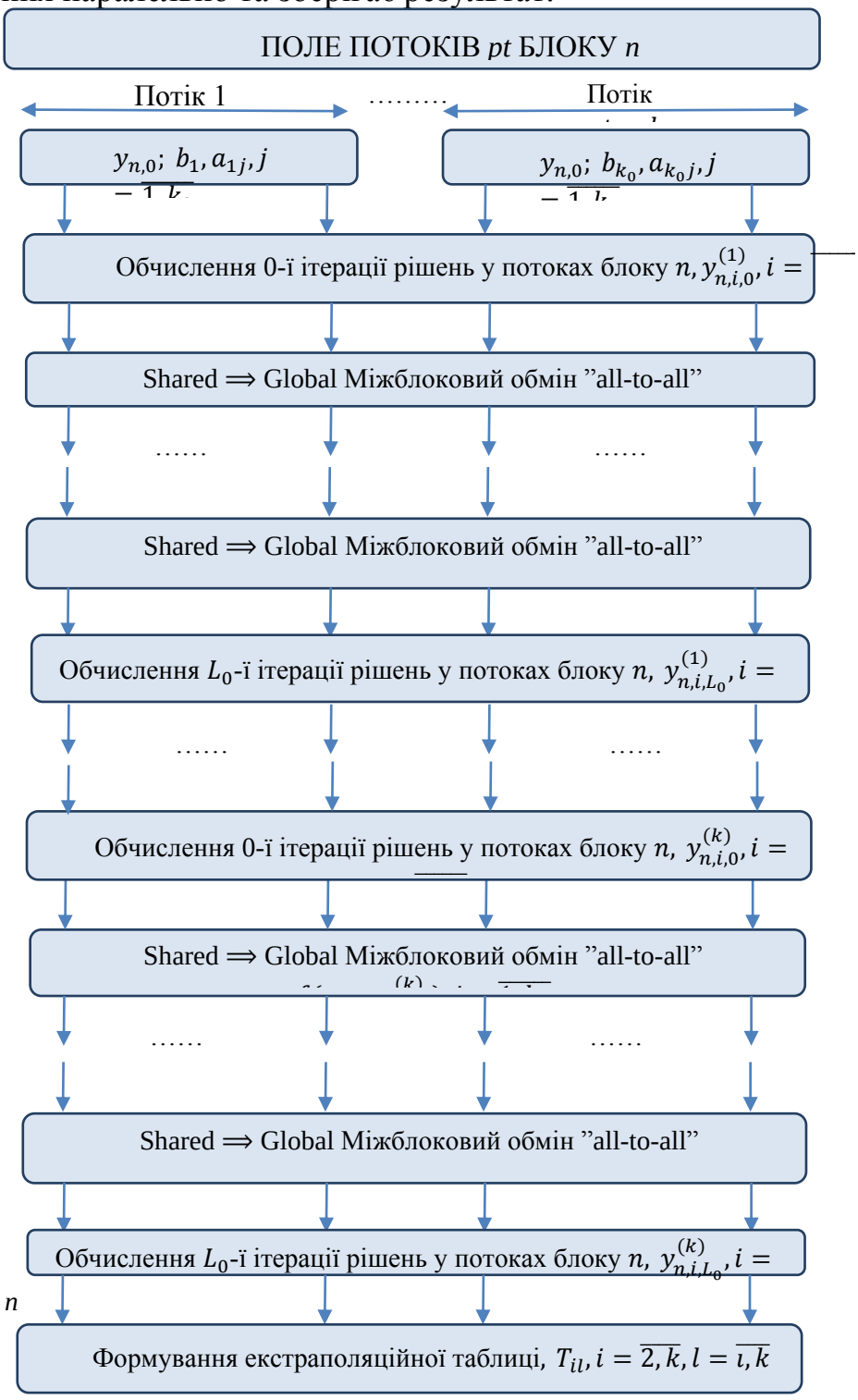


Рисунок 2 – Паралельний алгоритм технології локальної екстраполяції на основі однокрокового k_0 – точкового методу для ЗДР

Розроблений програмний продукт поєднує в собі модуль генерації системи ЗДР та паралельну реалізацію екстраполяційного алгоритму для розв'язання таких систем. Розв'язання СЗДР відбувається із використанням екстраполяції Річардсона, у якості опорного використано метод Фельберга 7(8). Початковий розмір таблиці екстраполяції - 5, але для отримання даних що до доречності підвищення цього параметра, також ведеться розрахунок значень для 6-го та 7-го рядка таблиці.

Керування кроком інтегрування здійснюється на підставі оцінювання масштабів відмінностей результатів, отриманих на дійсному та попередньому рівнях екстраполяції. Загалом ідея полягає в тому, що коли на новому рівні отримуємо значення, яке істотно відрізняється від попереднього, - треба продовжувати уточнювати розв'язання. Для цього потрібно збільшувати кількість рядків таблиці екстраполяції, або зменшувати крок інтегрування, або виконувати обидва прийоми.

3. ЕКСПЕРИМЕНАЛЬНЕ ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ЛЕР НА БАЗІ ТЕХНОЛОГІЇ CUDA

Для дослідження ефективності розроблених паралельних алгоритмів та їх програмних додатків було проведено чисельний експеримент, що дозволив отримати наступні динамічні характеристики: час паралельного виконання, загальні накладні витрати на паралелізм, прискорення та ефективність. Розроблене програмне забезпечення реалізовано для використання ЛЕР на базі однокрокових опорних методів таких, як явні багатостадійні симетричні та неявні блокові або багатоточкові методи. Для генерації сіток інтегрування застосовувалися гармонійний ряд та парна послідовність Ромберга [7].

Для дослідження залежності динамічних характеристик паралельних алгоритмів від розмірності СЗДР, числа використаних паралельних елементів, порядку реалізованого методу і інших параметрів була проведена серія експериментів. При цьому для графічних процесорів використовувалася відеокарта NVIDIA GeForce GT 610 (кількість ядр CUDA - 48). Приклади експериментальних залежностей динамічних характеристик, отриманих в результаті виконання програмних додатків, що реалізують ЛЕР, з використанням технологій CUDA та для порівняння MPI, наведено на рисунках 3-4.

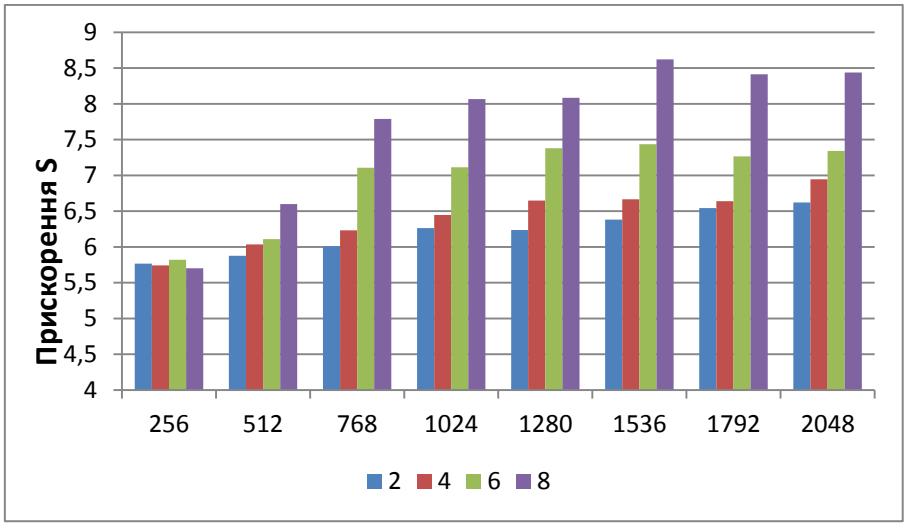


Рисунок 3 – Прискорення паралельного алгоритму ЛЕР від порядку опорного методу та розміру СЗДР для технології CUDA

Проведені експерименти свідчать про те, що використання технології CUDA є досить ефективним для вирішення СЗДР великої розмірності. Коефіцієнт прискорення, одержуваний при реалізації паралельних алгоритмів, зростає з ростом порядку методу і розмірності системи. Використання великого числа потоків, що виконуються паралельно, дає високі показники навіть на невеликих розмірах правих частин і низьких порядках методу, тим самим демонструючи в цих умовах перевагу технології CUDA над технологією MPI.

У силу обмеженості пам'яті, що в CUDA і кількості створюваних потоків на блоках, коефіцієнти прискорення ростуть не достатньо швидко, а при подальшому збільшенні кількості рівнянь в системі можуть навіть знижуватися. Однак і за таких умов алгоритми, реалізовані за допомогою технології CUDA, мають явну перевагу перед алгоритмами, що використовують тільки засоби пакета MPI. Таким чином, технологія CUDA з використанням пам'яті, що розділяється, є ефективним рішенням для інтегрування жорстких задач Коші для систем звичайних диференціальних рівнянь великої розмірності. Отримані дані вказують на те, що із зростанням кількості обчислювальних вузлів прискорення зростає, але темпи цього зростання скорочуються, ефективність у такому разі знижується.

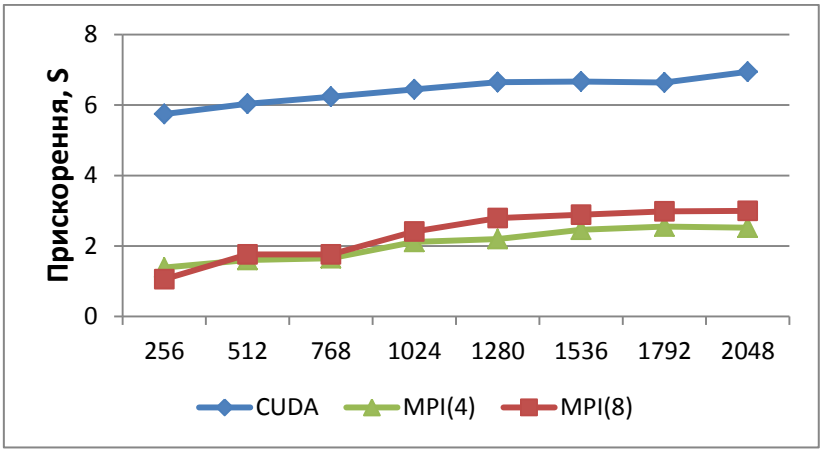


Рисунок 4 – Прискорення паралельних алгоритмів ЛЕР при опорних методах 4-го порядку для технологій CUDA та MPI

Також треба зазначити, що зростання прискорення є обмеженим через неможливість постійного збільшення розмірності таблиці екстраполяції. Коли максимальна розмірність таблиці екстраполяції стане менше за число наявних обчислювальних модулів, частина вузлів буде постійно простоювати, прискорення зростати не буде, а ефективність буде знижуватись все швидше.

4. ВИСНОВКИ

Досліджено підвищення ефективності реалізації технології локальної екстраполяції Річардсона при вирішенні задачі Коші для СЗДР. Встановлено, що найменш трудомісткими є обчислювальні схеми з симетричними опорними методами малих порядків точності у поєднанні з парними послідовностями чисел для генерації сіток інтегрування.

Побудовано паралельні алгоритми розв'язання нелінійної задачі Коші з контролем покрокової похибки на основі технології локальної екстраполяції Річардсона для однокрокових явних багатостадійних та неявних блокових опорних методів. Оцінено ефективність отриманих паралельних алгоритмів на основі рівномірного, пропорційного та комбінаційного способу розподілу даних по процесорам.

Розроблені обчислювальні схеми відображення отриманих паралельних алгоритмів на структури паралельних обчислювальних систем різних архітектур із розподіленою пам'яттю з наступними топологіями міжпроцесорних зв'язків: лінійка/кільце, решітка/тор, гіперкуб та на графічні процесори з використанням технології CUDA. Досліджено ефективність отриманих топологічних відображень в

залежності від типу архітектури паралельної ОС, різних топологій і комунікаційних констант, а також розмірності початкової задачі, складності її правої частини і унікальних характеристик методу. Визначено пріоритетні галузі застосування розроблених паралельних методів рішення динамічних задач у поєднанні з паралельною архітектурою. переваги явних схем, заснованих на технології локальної екстраполяції, проявляються при пошуку високоточних рішень і для складних правих частин.

ЛІТЕРАТУРА:

[1] **A. V. Boreskov, A. A. Charlamov.** Basics of working with CUDA technology. M.: Press, 2010. 232 p.

[2] NVIDIA CUDA – Non-graphical calculations on graphic processors [Electronic resource] // IXBT. – Access mode: http://www.ixbt.com/video3/cuda-1.shtml

[3] Programming Guide: CUDA Toolkit Documentation [Electronic resource] // NVIDIA. – Access mode: http://docs.nvidia.com/cuda/cuda-c-programming-guide

[4] **L. P. Feldman, I. A. Nazarova.** Parallel one-step methods for the numerical solution of the Cauchy's problem. *Donetsk National Technical University, Donetsk*, 2011, 185p.

[5] **L. P. Feldman, I. A. Nazarova.** Modern parallel methods for the numerical solution of the Cauchy's problem, *Donetsk National Technical University, Donetsk*, 2013, 206p.

[6] **L. P. Feldman, I. A. Nazarova, Y. A. Grizadubova**. Ефективність паралельної реалізації екстраполяційних методів чисельного розв'язання СЗДР на графічних процесорах. *Scientific papers of Donetsk National Technical University. Series: Informatics, Cybernetics and Computer Science №2, Pokrovsk,* 2016, P. 37-44.

[7] **I.A. Nazarova, Y.A. Grizadubova**. Чисельне рішення жорстких завдань Коші на основі блокових однокрокових методів з використанням технології CUDA// *X міжнародна науково-практична студентська конференція магістрантів (05-08 квітня 2016 року) – Харків: НТУ «ХПІ»*, 2016 – С. 107-108.

[8] **M. T. Chu, H. Hamilton**. Parallel Solution of ODE's by Multiblock Methods // *Journal on Scientific and Statistical Computing, vol. 8*, 1987, P. 342-353.

[9] **E. Hairer, S. P. Nørsett, G.Wanner**. Solving Ordinary Differential Equations I. Nonstiff Problems. *Springer*, 2009, 685p.

Отримано 20.06.2017

УДК 004.912

ПРОГРАМНА СИСТЕМА ДЛЯ ВИВЧЕННЯ ЛЕКСИКИ АНГЛІЙСЬКОЇ МОВИ

Ануфрієв П. О.
асистент кафедри прикладної математики та інформатики факультету комп'ютерних наук та технології Донецького національного технічного університету

Останнім часом все більше і більше педагогів-практиків розуміють, що використання інформаційних комунікаційних технологій (ІКТ) в навчальному процесі значно підвищує ефективність засвоєння матеріалу учнями. Застосування ІКТ під час занять дає можливість в будь-який момент витягти з пам'яті і використовувати мовленнєвий матеріал, що відноситься до різних розділів досліджуваної іноземної мови, проводити поглиблене структурування даного матеріалу, розташовувати його в будь-якій прийнятній формі, використовувати колір, анімацію, графічні та звукові можливості, регулювати динамічну послідовність їх пред'явлення. Більш адекватна форма презентації мовних знань дає позитивний психологічний ефект, в результаті забезпечується краще розуміння мовного матеріалу [1].

Крім цього, обговорюючи основні проблеми освоєння лексичного складу мови, багато дослідників загострюють увагу на питанні як, по якими ознаками можна судити про те, що слово засвоєно. Це питання безпосередньо пов'язане з тим, що ми розумінням під словом і під його значенням. Якщо підійти до питання про лексичному складі мови з позицій психолінгвістики, то одним з найважливіших питань постане питання про співвідношення слова, значення і поняття [2]. Якісно новий етап дослідження лексичного складу мови в психолінгвістиці пов'язаний з роботами Л. С. Виготського і його послідовних. Тут в психолінгвістичному відношенні одиницею аналізу стає не слово, а значення [2].

Звертаючи на це увагу, було вирішено створити програмну систему, яка вирішує наступні задачі:

- зосередити увагу саме на вивчення слів за допомогою її значення;
- забезпечити методичну складову вивчення англійської мови;
- використовувати сучасні засоби для аналізу і порівняння зі створюваною системою, використовувати сучасну англійську мову;

- забезпечити зручність вивчення мови за рахунок зручного й сучасного користувальницького інтерфейсу, швидкого доступу до тлумачного словника;
- забезпечити одночасну роботу з текстом та тлумачним словником;
- забезпечити можливість до того, як користувач прочитає текст, знайти незнайомі слова.
- забезпечити створення користувацького словника, та системи вправ, для вивчення.

На момент презентації програмної системи було вирішено всі задачі, крім останньої. Було застосовано сучасні методики та засоби навчання, наприклад «Словник сучасної англійської мови Longman» (Longman Dictionary of Contemporary English) [3]. Концепцію програми було порівняно з найпопулярнішими існуючими програмами для вивчення мови та виділено шляхи її розробки.

На рисунку 1 показано, як виглядає інтерфейс програми під час роботи та знаходження значення вибраного слова. Також на рисунку можна побачити, що програма надає інформацію щодо використання слова, приклади використання, значення слова в різних частинах мови, транскрипцію, та доступ до аудіозапису слова.

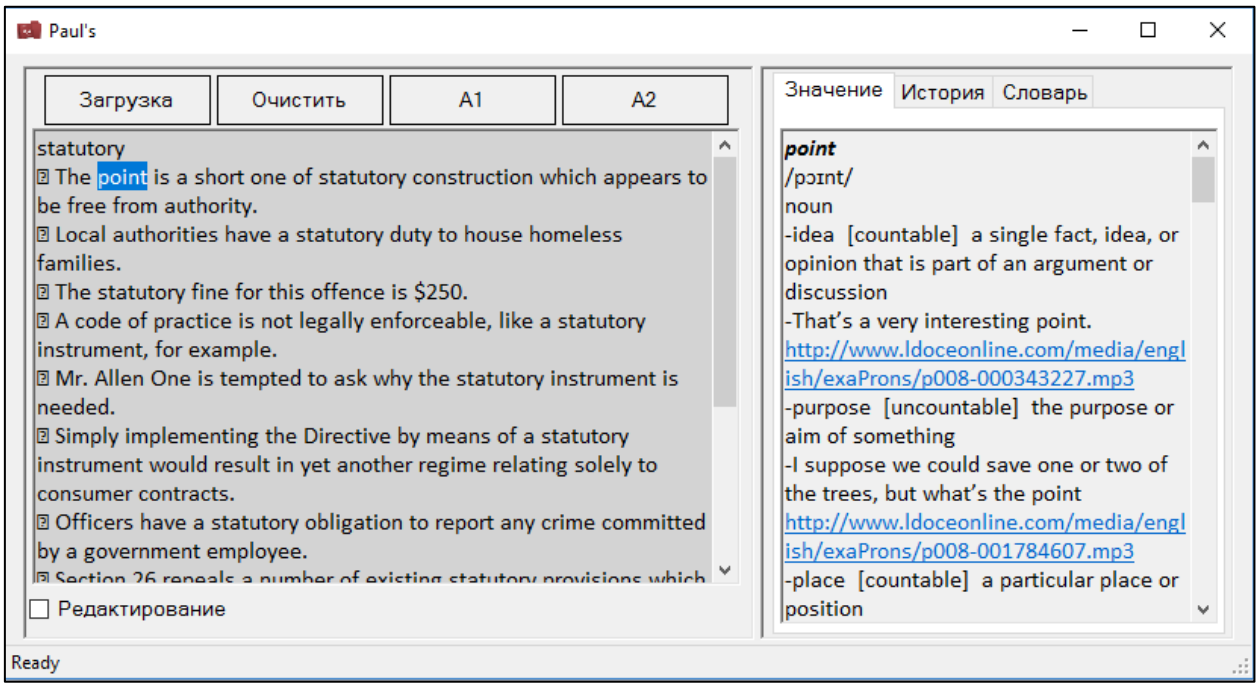


Рисунок 1 – Робота програми під час вибору слова

Програмна система була створена у середовищі розробки Visual Studio. Програмний код реалізовано мовою програмування C#. Проектування програми було розроблено за допомогою мови UML.

Наступними задачами поданою програмної системи можуть бути пошук конкретного значення слова звертаючи увагу на місце слова у контексті, завершити розробку користувацького словника та словника незнайомих слів, історію використання слів в системі. Також створити базу даних за допомогою мови SQL.

ЛІТЕРАТУРА

1. А. А. Бордашова, Н. Н. Нечаев| Проблемы освоения лексики иностранного языка.
2. Расулова М. Ж. Использование информационно- коммуникационных технологий на уроках английского языка [Текст] // Теория и практика образования в современном мире: материалы II Междунар. науч. конф. (г. Санкт-Петербург, ноябрь 2012 г.). — СПб.: Реноме, 2012. — С. 91-92.
3. Longman Dictionary of Contemporary English [Електронний ресурс] : [Інтернет – сайт] Режим доступу: http://www.ldoceonline.com/ Назва з екрану - Дата звернення 07.09.2017 – Мова англ.

УДК 004.92

МЕТОД ПРОГНОЗУ ПРИБУТКУ ІНТЕРНЕТ МАГАЗИНУ НА ОСНОВІ ПОВНОЗВ'ЯЗНОЇ ДОВГОКОРОТКОЧАСНОЇ ПАМ'ЯТІ

Є.Є. Федоров, д.т.н., доц, В.О. Патрушев, ст.викл.,
О.І. Патрушева, ас.
Донецький національний технічний університет

У статті розглянуті і проаналізовані існуючі методи прогнозу формування прибутку. Виходячи з основних переваг та недоліків, розроблений і реалізований нейросетевой метод прогнозу прибутку функціонуючого інтернет-магазину. В основу методу закладена нейронна мережа FC-LSTM, архітектура якої визначена на основі проведених експериментів. Критерієм вибору архітектури було мінімальне значення MSE. Для оцінки ефективності запропонованого методу було проведено чисельні дослідження, які доводять ефективність обраної мережі та її архітектури.

Ключові слова: прогноз, прибуток, господарська діяльність, нейронна мережа, ідентифікація структури і параметрів мережі, середньоквадратична помилка.

ЗАГАЛЬНА ПОСТАНОВКА ПРОБЛЕМИ.

В даний час однією з найважливіших проблем, з якою стикаються сучасні інтернет магазини є відсутність вбудованих механізмів, які дозволяють виконувати аналіз і подальше прогнозування результатів господарської діяльності, спрямованих на підвищення прибутку. Це призводить до того, що господарська діяльність власника інтернет магазину, спрямована на збільшення прибутку, може бути неефективною. Тому розробка способів прогнозу прибутку інтернет магазину є актуальною.

ПОСТАНОВКА ЗАВДАННЯ ДОСЛІДЖЕННЯ.

Метою роботи є розробка методу дослідження процесу формування прибутку інтернет-магазину. Для досягнення мети були поставлені і вирішені наступні завдання:

- провести аналіз існуючих методів прогнозу;
- вибрати штучну нейронну мережу прогнозу;
- визначити структури моделі штучної нейронної мережі;
- вибрати критерій оцінки ефективності нейросетевой моделі прогнозу;
- навчити моделі штучної нейронної мережі;

- виконати чисельні дослідження.

РЕЗУЛЬТАТИ РОЗРОБКИ ТА ДОСЛІДЖЕНЬ.

На сьогоднішній день в якості інструменту для прогнозу відомо безліч підходів, серед яких можна виділити:

- регресивні і авторегресійні методи прогнозування;
- методи, засновані на експоненційному згладжуванні;
- методи на базі ланцюгів Маркова;
- на базі классификационно-регресійних дерев;
- нейромережеві методи прогнозу.

В роботі [5] наведено порівняльну характеристику перерахованих способів.

Оскільки використання нейронних мереж при прогнозуванні дає відчутну перевагу, яке полягає в тому, що: взаємозв'язки між факторами досліджуються на готових моделях; не потрібні ніякі припущення щодо розподілу факторів; апріорна інформація про фактори може бути відсутнім; вихідні дані можуть сильно корелювати, бути неповними або зашумленими; можливий аналіз систем з високою ступенем не лінійності; швидка розробка моделі; висока адаптивність; можливий аналіз систем з великою кількістю чинників; не вимагається повний перебір всіх можливих моделей; можливий аналіз систем з неоднорідними факторами, в статті буде використовуватися нейросетевой спосіб прогнозу.

Оскільки фактори, що впливають на отримання прибутку змінюються як в короткостроковій, так і довгостроковій перспективі, то прогноз здійснюється на основі динамічних (з тимчасовою затримкою) штучних нейронних мереж. В якості таких мереж найчастіше використовуються:

- нейросеть Джордона (JNN) [6], яка є динамічною рекуррентной двошарової мережею і побудована на базі MLP.
- нейросеть Елмана (ENN) або проста рекуррентная мережу (SRN) [6], яка є динамічною рекуррентной двошарової мережею і побудована на базі MLP.
- рекурентний багатошаровий персептрон (RMLP) [7], який є динамічною рекуррентной багатошарової мережею і побудований на базі MLP.
- модель нелінійної авторегресії (NAR) [8], яка є динамічною нерекуррентной двошарової мережею і побудована на базі MLP.
- модель нелінійної авторегресії-ковзного середнього (NARMA) [8], яка є динамічною рекуррентной двошарової мережею і побудована на базі MLP.

- полносвязная долгократкосрочная пам'ять (FC-LSTM) [9, 10], яка є рекуррентной двошарової мережею і побудована на базі MLP.

У табл. 1 наведені порівняльні характеристики динамічних мереж. Кількість нейронів вхідного шару відповідає кількості контрольованих параметрів. Дослідження проводилися для одного показника – прибуток інтернет-магазину. Це відповідає одному нейрона вхідного шару.

Таблиця 1 - Результати експериментальних досліджень нейронних мереж

Мережа / Критерій	RLMP	JNN	ENN (SRN)	NAR	NARMA	FC-LSTM
Наявність зворотного зв'язку	+	+	-	-	+	+
Затримка у вхідному шарі	-	-	-	+	+	+
Точність прогнозу	90	80	85	88	93	95

Оскільки FC-LSTM дає найбільшу точність прогнозу, ця мережа вибирається для вирішення завдань прогнозування.

В даній статті запропоновано авторський варіант FC-LSTM, який передбачає тимчасові затримки у вхідному шарі. Структура FC-LSTM з тимчасовими затримками у вхідному шарі зображена на рис.1.

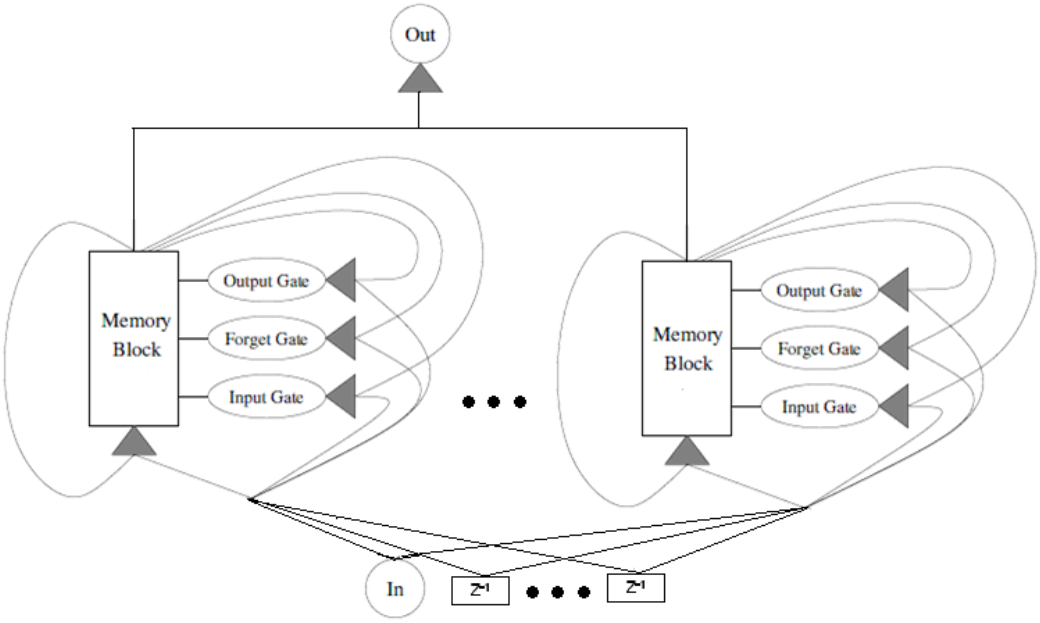


Рисунок 1 - Повнозв'язна довгокороткочасна пам'ять (FC-LSTM) з часовими затримками в вхідному шарі

В якості прихованого нейрона використовується блок пам'яті (Memory Block), який містить одну або більше комірок, і пов'язаний з вхідним шлюзом (Input Gate), вихідним шлюзом (Output Gate) і шлюзом забування (Forget Gate), які відповідають КИХ-фільтрів. Зв'язки між осередком і шлюзами дозволять шлюзів враховувати стан осередку. Вихідний сигнал осередки надходить по зворотного зв'язку назад в блок пам'яті.

$$y_j^{gin}(n) = f\left(\sum_{i=0}^{M^{(0)}} w_{ij}^{in-gin} y^{in}(n-(i+1)) + \sum_{v=1}^{S_j} w_{jvj}^{s-gin} s_{jv}^{c}(n-1) + \sum_{v=1}^{S_j} w_{jvj}^{c-gin} y_{jv}^{c}(n-1)\right),$$

$j \in \overline{1, N^{(1)}}$,

$$y_j^{g\varphi}(n) = f\left(\sum_{i=0}^{M^{(0)}} w_{ij}^{in-g\varphi} y^{in}(n-(i+1)) + \sum_{v=1}^{S_j} w_{jvj}^{s-g\varphi} s_{jv}^{c}(n-1) + \sum_{v=1}^{S_j} w_{jvj}^{c-g\varphi} y_{jv}^{c}(n-1)\right),$$

$j \in \overline{1, N^{(1)}}$,

$$net_{jv}^{c}(n) = \sum_{i=0}^{M^{(0)}} w_{ijv}^{in-s} y^{in}(n-(i+1)) + \sum_{z=1}^{S_j} w_{jzjv}^{c-s} y_{jz}^{c}(n-1),$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$s_{jv}^{c}(n) = y_j^{g\varphi}(n) s_{jv}^{c}(n-1) + y_j^{gin}(n) g(net_{jv}^{c}(n)),$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$y_j^{gout}(n) = f\left(\begin{array}{l}\sum\limits_{i=0}^{M^{(0)}} w_{ij}^{in-gout} y^{in}(n-(i+1)) + \sum\limits_{v=1}^{S_j} w_{jvj}^{s-gout} s_{jv}^{c}(n-1) + \\ + \sum\limits_{v=1}^{S_j} w_{jvj}^{c-gout} y_{jv}^{c}(n-1)\end{array}\right),$$

$j \in \overline{1, N^{(1)}}$,

$$y_{jv}^{c}(n) = y_j^{gout}(n) h(s_{jv}^{c}(n)),\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}},$$

$$y^{out}(n) = f\left(\sum_{i=1}^{N^{(1)}} \sum_{v=1}^{S_i} w_{iv}^{c-out} y_{iv}^{c}(n)\right).$$

де $M^{(0)}$ - затримка у вхідному шарі; $N^{(1)}$ - кількість блоків пам'яті; S_j - кількість осередків в j-му блоці пам'яті; $w_{ij}^{in-gin}, w_{ij}^{in-g\varphi}, w_{ij}^{in-gout}$, - вага зв'язку від i-го вхідного нейрона в момент часу n - (i +1) до вхідного, вихідного і забування шлюзів j-го блоку пам'яті в момент часу n; w_{ijv}^{in-s} - вага зв'язку від i-го вхідного нейрона в момент часу n - (i +1) до v-й осередку j-го блоку пам'яті в момент часу n; $w_{jvj}^{s-gin}, w_{jvj}^{s-g\varphi}, w_{jvj}^{s-gout}, w_{jvj}^{c-gin}, w_{jvj}^{c-g\varphi}, w_{jvj}^{c-gout}$, w_{jzjv}^{c-s} - вага зв'язку від v-й осередку j-го блоку пам'яті в момент часу n - (i +1) до вхідного, вихідного і забування шлюзів j-го блоку пам'яті в момент часу n; w_{iv}^{c-out} - вага зв'язку від v-й осередку i-го блоку пам'яті в момент часу n - (i +1) до v-й осередку i-го блоку пам'яті в момент часу n; $y_j^{gin}(n)$ - вихід вхідного шлюзу j-го блоку пам'яті; $y_j^{gout}(n)$ - вихід вихідного шлюзу j-го блоку пам'яті; $y_j^{g\varphi}(n)$ - вихід шлюзу забування j-го блоку пам'яті; $y_{jv}^{c}(n)$ - вихід v-й осередку j-го блоку пам'яті; $y^{in}(n)$ - вихід вхідного нейрона; $y^{out}(n)$ - вихід вихідного нейрона; $s_{jv}^{c}(n)$ - стан v-й осередку j-го блоку пам'яті; f - функція активації нейронів.

Для визначення структури моделі FC-LSTM (визначення кількості блоків пам'яті) був проведений ряд експериментів, результати яких представлені на рис. 2.

В якості вхідних даних для навчання FC-LSTM були використана вибірка значень по господарської діяльності існуючого інтернет-магазину за 2016 рік. Критерієм вибору структури моделі мережі було мінімальна середньоквадратична помилка прогнозу. Як видно з рис. 2, зі збільшенням кількості одноячеечних блоків пам'яті значення помилки зменшується. Для прогнозу продажів і отримання подальшої прибутку досить використовувати 5 одноячеечних блоків пам'яті, оскільки при подальшому збільшенні кількості одноячеечних блоків пам'яті зміна значення помилки незначно.

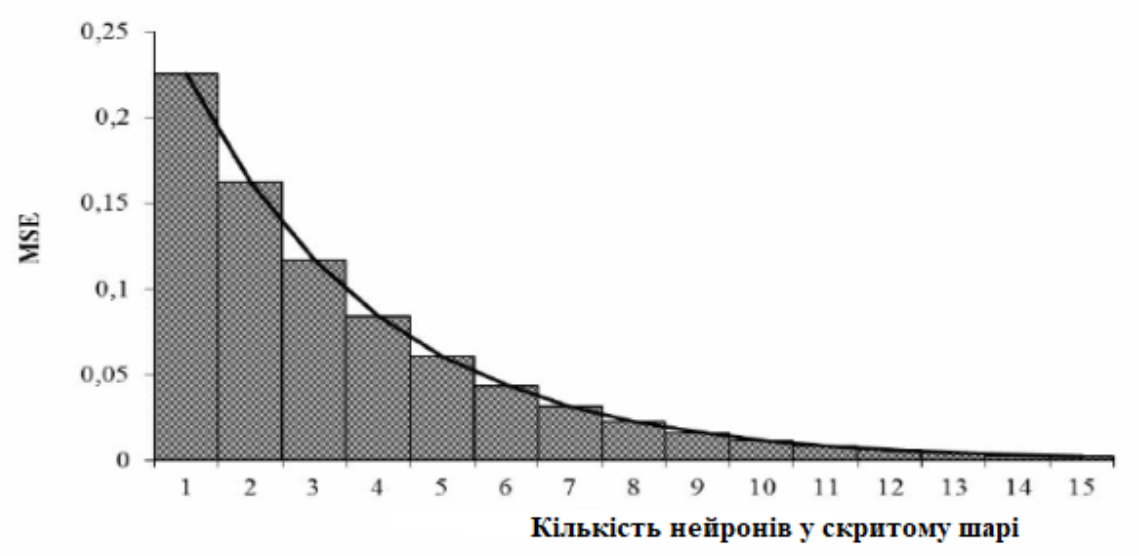


Рисунок 2 – Гістограма залежності значення середньоквадратичної помилки від кількості нейронів у скритому шарі

В роботі для навчання моделі FC-LSTM обраний критерій адекватності моделі, який означає вибір таких значень параметрів $W = \{w_{ij}^{in-gin},\ w_{ij}^{in-g\varphi},\ w_{ij}^{in-gout},\ w_{ijv}^{in-s},\ w_{jvj}^{s-gin},\ w_{jvj}^{s-g\varphi},\ w_{jvj}^{s-gout},\ w_{jvj}^{c-gin},\ w_{jvj}^{c-g\varphi},\ w_{jvj}^{c-gout},\ w_{jzjv}^{c-s},\ w_{iv}^{c-out}\}$, які доставляють мінімум середньоквадратичної помилки (різниці виходу за моделлю і бажаного виходу):

$$F = \frac{1}{P}\sum_{p=1}^{P}(y_p - d_p)^2 \to \min_W \quad (3)$$

Навчання моделі FC-LSTM підпорядковане критерієм (3), для чого в статті пропонується комбінація алгоритмів зворотного поширення в часі (BPTT) і рекуррентного навчання в реальному часі (RTRL), яка полягає в наступному

1. Номер ітерації навчання, ініціалізація за допомогою рівномірного розподілу на інтервалі (0,1) або [-0.5, 0.5] ваг $w_{ij}^{in-gin}(n)$, $w_{ij}^{in-g\varphi}(n)$, $w_{ij}^{in-gout}(n)$, $w_{ijv}^{in-s}(n)$, $i \in \overline{0, M^{(0)}}$, $j \in \overline{1, N^{(1)}}$, $v \in \overline{1, S_j}$, $w_{jvj}^{s-gin}(n)$, $w_{jvj}^{s-g\varphi}(n)$, $w_{jvj}^{s-gout}(n)$, $w_{jvj}^{c-gin}(n)$, $w_{jvj}^{c-g\varphi}(n)$, $w_{jvj}^{c-gout}(n)$, $w_{jzjv}^{c-s}(n)$, $j \in \overline{1, N^{(1)}}$, $z \in \overline{1, S_j}$, $v \in \overline{1, S_j}$, $w_{iv}^{c-out}(n)$, $i \in \overline{1, N^{(1)}}$, $v \in \overline{1, S_j}$, где $M^{(0)}$ – затримка у вхідному шарі, $N^{(1)}$ – кількість блоків пам'яті, S_j – кількість осередків у j-м блоці пам'яті.

2. Задається навчальна множина $\{(x_\mu, d_\mu) \mid x_\mu \in R, d_\mu \in R\}$, $\mu \in \overline{1, P}$, де x_μ – μ-е вхідне значення, що навчає, d_μ – μ-е вихідне значення, що навчає, P – потужність навчальної множини. Номер поточної пари з навчальної множини.

3. Початкове обчислення вихідного сигналу осередку $y^{in}(n-(i+1)) = 0$, $i \in \overline{1, M^{(0)}}$,

$$s_{jv}^{c}(n-1) = 0,\ y_{jv}^{c}(n-1) = 0,\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}}.$$

4. Обчислення вихідного сигналу для кожного шару (прямий хід)

$$y^{in}(n-1) = x_{\mu-1},$$

$$y_j^{gin}(n) = f(net_j^{gin}(n)),\ j \in \overline{1, N^{(1)}},$$

$$net_j^{gin}(n) = \sum_{i=0}^{M^{(0)}} w_{ij}^{in-gin}(n) y^{in}(n-(i+1)) + \sum_{v=1}^{S_j} w_{jvj}^{s-gin}(n) s_{jv}^{c}(n-1) +$$

$$+ \sum_{v=1}^{S_j} w_{jvj}^{c-gin}(n) y_{jv}^{c}(n-1),$$

$$f(s)=\frac{1}{1+e^{-s}},$$

$$y_j^{g\varphi}(n)=f(net_j^{g\varphi}(n)),\ j\in\overline{1,N^{(1)}},$$

$$net_j^{g\varphi}(n)=\sum_{i=0}^{N^{(0)}}w_{ij}^{in-g\varphi}(n)y^{in}(n-(i+1))+\sum_{v=1}^{S_j}w_{jvj}^{s-g\varphi}(n)s_{jv}^{c}(n-1)+$$

$$+\sum_{v=1}^{S_j}w_{jvj}^{c-g\varphi}(n)y_{jv}^{c}(n-1),$$

$$f(s)=\frac{1}{1+e^{-s}},$$

$$net_{jv}^{c}(n)=\sum_{i=0}^{M^{(0)}}w_{ijv}^{in-s}(n)y^{in}(n-(i+1))$$

$$+\sum_{z=1}^{S_j}w_{jzjv}^{c-s}(n)y_{jz}^{c}(n-1),\ v\in\overline{1,S_j},\ j\in\overline{1,N^{(1)}},$$

$$s_{jv}^{c}(n)=y_j^{g\varphi}(n)s_{jv}^{c}(n-1)+y_j^{gin}(n)g(net_{jv}^{c}(n)),$$

$$v\in\overline{1,S_j},\ j\in\overline{1,N^{(1)}},$$

$$g(s)=2\tanh(s),$$

$$y_j^{gout}(n)=f(net_j^{gout}(n)),\ j\in\overline{1,N^{(1)}},$$

$$net_j^{gout}(n)=\sum_{i=0}^{M^{(0)}}w_{ij}^{in-gout}(n)y^{in}(n-(i+1))+\sum_{v=1}^{S_j}w_{jvj}^{s-gout}(n)s_{jv}^{c}(n-1)+$$

$$+\sum_{v=1}^{S_j}w_{jvj}^{c-gout}(n)y_{jv}^{c}(n-1),$$

$$f(s)=\frac{1}{1+e^{-s}},$$

$$y_{jv}^{c}(n)=y_j^{gout}(n)h(s_{jv}^{c}(n)),\ v\in\overline{1,S_j},\ j\in\overline{1,N^{(1)}},$$

$$y^{out}(n) = f(net^{out}(n)),$$

$$f(s) = \frac{1}{1+e^{-s}},\ net^{out}(n) = \sum_{i=1}^{N^{(1)}} \sum_{v=1}^{S_i} w_{iv}^{c-out}(n) y_{iv}^{c}(n-1).$$

5. Обчислення енергії помилки прогнозу нейромережі

$$E(n) = \frac{1}{2} e^2(n),\ e(n) = y^{out}(n) - d_{\mu-1}.$$

6. Налаштування синаптичних ваг на основі узагальненого дельта правила (зворотний хід)

$$w_{iv}^{c-out}(n+1) = w_{iv}^{c-out}(n) - \eta \frac{\partial E(n)}{\partial w_{iv}^{c-out}(n)},\ i \in \overline{1, N^{(1)}},\ v \in \overline{1, S_j},$$

$$w_{ij}^{in-gout}(n+1) = w_{ij}^{in-gout}(n) - \eta \frac{\partial E(n)}{\partial w_{ij}^{in-gout}(n)}, \qquad i \in \overline{1, N^{(0)}},$$

$j \in \overline{1, N^{(1)}}$,

$$w_{ijv}^{in-s}(n+1) = w_{ijv}^{in-s}(n) - \eta e_{jv}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{ijv}^{in-s}(n)}, \qquad i \in \overline{1, N^{(0)}},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{jzjv}^{c-s}(n+1) = w_{jzjv}^{c-s}(n) - \eta \sum_{z=1}^{S_j} e_{jz}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{jzjv}^{c-s}(n)}, \qquad z \in \overline{1, S_j},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{ij}^{in-gin}(n+1) = w_{ij}^{in-gin}(n) - \eta \sum_{v=1}^{S_j} e_{jv}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-gin}(n)},$$

$i \in \overline{1, N^{(0)}}$, $v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{ij}^{in-g\varphi}(n+1) = w_{ij}^{in-g\varphi}(n) - \eta \sum_{v=1}^{S_j} e_{jv}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-g\varphi}(n)},$$

$i \in \overline{1, N^{(0)}}$, $v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{jvj}^{s-gout}(n+1) = w_{jvj}^{s-gout}(n) - \eta \frac{\partial E(n)}{\partial w_{ivj}^{s-gout}(n)},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{jvj}^{s-gin}(n+1) = w_{jvj}^{s-gin}(n) - \eta \sum_{z=1}^{S_j} e_{jz}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{s-gin}(n)},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{jvj}^{s-g\varphi}(n+1) = w_{jvj}^{s-g\varphi}(n) - \eta \sum_{z=1}^{S_j} e_{jz}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{s-g\varphi}(n)},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{jvj}^{c-gout}(n+1) = w_{jvj}^{c-gout}(n) - \eta \frac{\partial E(n)}{\partial w_{ivj}^{c-gout}(n)},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{jvj}^{c-gin}(n+1) = w_{jvj}^{c-gin}(n) - \eta \sum_{z=1}^{S_j} e_{jz}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{c-gin}(n)},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

$$w_{jvj}^{c-g\varphi}(n+1) = w_{jvj}^{c-g\varphi}(n) - \eta \sum_{z=1}^{S_j} e_{jz}^{c}(n) \frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{c-g\varphi}(n)},$$

$v \in \overline{1, S_j}$, $j \in \overline{1, N^{(1)}}$,

где η – параметр, що визначає швидкість навчання (при великому навчання відбувається швидше, але збільшується небезпека одержати невірне рішення), $0 < \eta < 1$.

$$\frac{\partial E(n)}{\partial w_{iv}^{c-out}(n)} = y_{iv}^{c}(n-1)\delta^{out}(n),$$

$$\frac{\partial E(n)}{\partial w_{ij}^{in-gout}(n)} = y_{i}^{in}(n-1)\delta_{j}^{gout}(n),$$

$$\frac{\partial E(n)}{\partial w_{jvj}^{s-gout}(n)} = s_{jv}^{c}(n-1)\delta_{j}^{gout}(n),$$

$$\frac{\partial E(n)}{\partial w_{jvj}^{c-gout}(n)} = y_{jv}^{c}(n-1)\delta_{j}^{gout}(n),$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{ijv}^{in-s}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{ijv}^{in-s}(n-1)} y_{j}^{g\varphi}(n) + & n > 2 \\ + y^{in}(n-(j+1)) y_{j}^{gin}(n) g'(net_{jv}^{c}(n)), & \\ y^{in}(n-(j+1)) y_{j}^{gin}(n) g'(net_{jv}^{c}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{izjv}^{c-s}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{ijv}^{c-s}(n-1)} y_{j}^{g\varphi}(n) + y_{iz}^{c}(n-1) y_{j}^{gin}(n) g'(net_{jv}^{c}(n)), & n > 2 \\ y_{z}^{c}(n-1) y_{j}^{gin}(n) g'(net_{jv}^{c}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-g\varphi}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{ij}^{in-g\varphi}(n-1)} y_{j}^{\varphi}(n) + & n > 2 \\ + y^{in}(n-(j+1)) s_{jv}^{c}(n-1) f'(net_{j}^{\varphi}(n)), & \\ y^{in}(n-(j+1)) s_{jv}^{c}(n-1) f'(net_{j}^{\varphi}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{c-g\varphi}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{jzj}^{c-g\varphi}(n-1)} y_{j}^{\varphi}(n) + & n > 2 \\ + y_{jz}^{c}(n-1) s_{jv}^{c}(n-1) f'(net_{j}^{\varphi}(n)), & \\ y_{jz}^{c}(n-1) s_{jv}^{c}(n-1) f'(net_{j}^{\varphi}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{s-g\varphi}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{jzj}^{s-g\varphi}(n-1)} y_{j}^{\varphi}(n) + & n > 2 \\ + s_{jz}^{c}(n-1) s_{jv}^{c}(n-1) f'(net_{j}^{\varphi}(n)), & \\ s_{jz}^{c}(n-1) s_{jv}^{c}(n-1) f'(net_{j}^{\varphi}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{ij}^{in-gin}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{ij}^{in-gin}(n-1)} y_j^{g\varphi}(n) + & n > 2 \\ + y^{in}(n-(j+1)) g(net_{jv}^{c}(n)) f'(net_j^{gin}(n)), & \\ y^{in}(n-(j+1)) g(net_{jv}^{c}(n)) f'(net_j^{gin}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{c-gin}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{izj}^{c-gin}(n-1)} y_j^{g\varphi}(n) + & n > 2 \\ + y_{jz}^{c}(n-1) g(net_{jv}^{c}(n)) f'(net_j^{gin}(n)), & \\ y_{jz}^{c}(n-1) g(net_{jv}^{c}(n)) f'(net_j^{gin}(n)), & n = 2 \end{cases},$$

$$\frac{\partial s_{jv}^{c}(n)}{\partial w_{jzj}^{s-gin}(n)} = \begin{cases} \dfrac{\partial s_{jv}^{c}(n-1)}{\partial w_{izj}^{s-gin}(n-1)} y_j^{g\varphi}(n) + & n > 2 \\ + s_{jz}^{c}(n-1) g(net_{jv}^{c}(n)) f'(net_j^{gin}(n)), & \\ s_{jz}^{c}(n-1) g(net_{jv}^{c}(n)) f'(net_j^{gin}(n)), & n = 2 \end{cases},$$

$$e_{jv}^{c}(n) = y_j^{gout}(n) h'(s_{jv}^{c}(n)) w_{jv}^{c-out}(n) \delta^{out}(n),$$

$$\delta^{out}(n) = f'(net^{out}(n))(y^{out}(n) - d_{\mu}),$$

$$\delta_j^{gout}(n) = f'(net_j^{gout}(n)) \sum_{v=1}^{S_j} h(s_{jv}^{c}(n)) w_{jv\backslash}^{c-out}(n) \delta^{out}(n).$$

7. Перевірка умови завершення

Якщо $(n-1) \bmod P > 0$, то $\mu = \mu + 1$, $n = n + 1$, перехід до 4.

Якщо $(n-1) \bmod P = 0$ та $\frac{1}{P} \sum_{s=1}^{P} E(n-P+s) > \varepsilon$, то $n = n + 1$, перехід до 2.

Якщо $(n-1) \bmod P = 0$ та $\frac{1}{P} \sum_{s=1}^{P} E(n-P+s) < \varepsilon$, то завершення.

ВИСНОВКИ.

1. Для вирішення проблеми підвищення точності прогнозу отримання прибутку інтернет-магазином були досліджені існуючі методи прогнозу прибутку підприємством. Дані дослідження показали, що на сьогоднішній день найбільш ефективним є використання штучних нейронних мереж.

2. Для підвищення якості прогнозу була обрана найбільш ефективна динамічна нейромережа і визначена структура її моделі. Проведені експерименти показали, що при десяти прихованих нейронах, значення середньоквадратичної помилки істотно не змінюється, і обрана мережа дає результати прогнозу з мінімальним відхиленням.

3. В якості алгоритму навчання була запропонована комбінація алгоритмів зворотного поширення в часі (BPTT) і рекуррентного навчання в реальному часі (RTRL). Це дозволило забезпечити високу точність прогнозу.

ЛІТЕРАТУРА

1. Чучуева І.А. Модель прогнозування часових рядів за вибіркою максимального подібності: дис. . ... кандидата технічних наук: 05.13.18 - математичне моделювання, чисельні методи та комплекси програм / Чучуева Ірина Олександрівна. - Москва, 2012. - 153 с.

2. Бодянський Є.В. Штучні нейронні мережі: архітектури, навчання, застосування / Є.В. Бодянський, О.Г. Руденко. - Харків: Телетех, 2004. - 159 с.

3. Haykin S. Neural networks / S. Haykin. - NY: Pearson Education, 1999. - p. 823.

4. Федоров Є.Є. Штучні нейронні мережі: монографія / О.Є. Федоров. - Красноармійськ: ДВНЗ «ДонНТУ», 2016. - 338 с.

5. Hochreiter S. Long short-term memory / S. Hochreiter, J. Schmidhuber // Technical Report FKI-207-95, Fakultat fur Informatik, Technische Universitat Munchen.

6. Gers F. Long Short-Term Memory in Recurrent Neural Networks / F. Gers // PhD thesis, Ecole Polytechnique Federale de Lausanne.

Отримано 30.06.2017

УДК 004.056.55

АЛГЕБРАЇЧНИЙ АНАЛІЗ ШИФРІВ, ОСНОВАНИХ НА КРИПТОГРАФІЧНИХ ПЕРЕТВОРЕННЯХ В АЛГЕБРІ ПОЛІВ ГАЛУА

В. О. Псьол, Н. О. Маслова
Державний вищий навчальний заклад «Донецький національний технічний університет»

Було проаналізовано стандарт шифрування AES, як базовий стандарт шифрів, основаних на криптографічних перетвореннях в правилах алгебри полів Галуа. Всі основні функції шифру було розглянуто та приведено до запису в правилах кінцевих полів, що дасть змогу в подальшому сформувати повноцінну алгебраїчну модель шифру.

Ключові слова: поля Галуа, дискретне логарифмування, криптоаналіз, криптографічні перетворення.

ВСТУП

Збереження конфіденційності інформації є однією з найбільш ключових проблем сучасного інформаційного світу. Для її вирішення було створено чимало шифрів. Серед інших вимог до таких шифрів можна виділити 3: простота, швидкодія, надійність. Певна річ, надійність шифру понад усе, а проте і дві інші вимоги важливі. Зі всієї множини існуючих шифрів всі 3 вимоги виконує шифр Rijndael, недарма він був прийнятий як державний стандарт шифрування AES в Сполучених Штатах Америки.

А проте, в його простоті криється, можливо, і його слабкість. Так, багато криптоаналітиків, а серед них і Брюс Шнайєр [1], і Алекс Бірюков [2], робили атаки на шифр. Ці атаки базувалися на лінійному криптоаналізі, і скажімо, показували швидкодію 2^{45} операцій для 10-раундового шифру. Однак, для повнораундового варіанту AES вони виявилися заслабкими [3]. Проте об'єднання таких атак із алгебраїчним підходом виглядає досить перспективним, адже в такому випадку вже функція заміни з використанням S-box буде не просто "чорною скринькою", а повноцінною математичною функцією, яку можна буде об'єднати із іншими та спростити.

І беручи до уваги, що об'єктом дослідження будуть являтися процеси захисту інформації, а предметом – алгоритми шифрування,

основані на полях Галуа, детально проаналізуємо шифр AES та приведемо всі його функції до вигляду алгебри полів Галуа.

АНАЛІЗ ФУНКЦІЙ AES

В алгоритмі шифрування AES використовуються 4 основні функції: SubBytes(), ShiftRows(), MixColumns() та AddRoundKey() [4]. Всі ці функції виконуються пораундово (10, 12 або 14 раундів в залежності від модифікації шифру) саме в такій послідовності. Розберемо їх по черзі.

Функція SubBytes() виконує нелінійну заміну байтів. Для цього формується таблиця нелінійних замін S-box. Використання цієї функції робить алгоритм нелінійним. Для того, щоб побудувати таку таблицю використовується 2 кроки. По-перше, для кожного із байтів береться зворотне число з поля Галуа $Gf(2^8)$, по-друге для кожного байта b, із яких складається S-box, виконується операція, представлена на формулі 1:

$$b_i' = b_i \oplus b_{(i+4)\ mod\ 8} \oplus b_{(i+5)\ mod\ 8} \oplus b_{(i+6)\ mod\ 8} \oplus b_{(i+7)\ mod\ 8} \oplus c_i, \quad (1)$$

де $0 \le i < 8$;

b_i – i-й біт байта b;

операція $\oplus$ – додавання по модулю 2;

c_i – i-й біт константи $\mathbf{c} = 63_{16} = 99_{10} = 01100011_2$.

Таким чином, із формули 1 бачимо, що забезпечується захист від простих алгебраїчних атак. Та більш детальної уваги потребує операція взяття зворотного числа з поля Галуа, іншими словами, операція розкладання многочлена, представленого в числовому вигляді, над полем $Gf(2^8)$.

Перш ніж говорити про саму функцію ShiftRows() необхідно сказати декілька слів про те, в якому форматі представляється сам блок даних, який поступає в шифрування та отримується на виході після цього. Такий блок представляється як певна структура даних, що містить 4 рядки інформації. Оскільки розмір блоку у нашому випадку має 128 біт, то очевидно, що в кожному із рядків буде рівно по 32 біти, тобто по 4 байти.

Під час виконання функції ShiftRows() буде здійснюватись здвиг рядків у блокові. Це вносить певну дифузію в шифр. Всі байти циклічно

здвигаються по-горизонталі, причому байти першого рядка не чіпаються взагалі (здвигаються на 0 позицій), байти другого рядку здвигаються на 1 позицію, третього – на 2, і четвертого – на 3. Таким чином після примінення процедури кожна колонка буде складатися із певної діагоналі (якщо порівнювати з блоком, який був до початку процедури).

Функція MixColumns() перемішує стовпці в кожному з блоків. Для цього приймається, що кожен із стовпців – це многочлен третього степеню. Потім кожен із таких многочленів перемножується по модулю (x^4+1) на фіксований многочлен $c(x)=(3x^3+x^2+x+2)$ в правилах алгебри кінечних полів. Як і в ShiftRows(), в MixColumns() використовується обратима трансформація, що вносить додаткову дифузію в шифр і ускладнює його криптоаналіз.

Функція AddRoundKey() додає до кожного блоку сам ключ. Додавання проводиться в правилах алгебри полів Галуа (XOR).

Кожен раундовий ключ, або і ключ взагалі повністю дорівнює по розміру блоку даних для шифрування. Це дає змогу примінити до кожного біту блоку операцію XOR із ключем (і-ий біт блоку $b_i = b_i \oplus k_i$). На виході, що очевидно, буде отримано блок тієї ж довжини, що дасть змогу знову приміняти до нього операції ShiftRows(), MixCollums() та AddRoundKey().

Серед усіх функцій ShiftRows() не виконує жодної зміни даних, а лише вносить в них дифузію, отже приводити її до правил алгебри немає сенсу. Приведемо функцію SubBytes(). Нехай байт стану позначимо як b, тоді:

$$b' \oplus \left[\left(\left((b' \times 2^4) mod\, 2^8\right) \times 2^4\right) \oplus \left((b' \times 2^4) div\, 2^8\right)\right] \oplus \ldots \qquad (2)$$

Зображення функції в правилах звичайної алгебри побайтно, як це видно із формули 2, виглядає громіздко, та збиттєво. Напишемо це перетворення за допомогою алгебри кінечних полів. Для цього виберемо поле, описавши його мультиплікативну групу. Очевидно, що поле має бути того ж розміру, що й базове, тобто $Gf(2^8)$. В той же час емпіричним шляхом було встановлено, що правилом заміни для описання такої операції буде $x^8=1$, тоді запишемо байт b' у вигляді многочлену $a_7x^7 \oplus a_6x^6 \oplus a_5x^5 \oplus a_4x^4 \oplus a_3x^3 \oplus a_2x^2 \oplus a_1x \oplus a_0$, де $a_0 - a_7$ – це, відповідно 0-7 біти байту b', при зчитуванні їх справа-наліво. Тоді формула 2 після спрощення прийме наступний вигляд:

$$b' \otimes (x^7+x^6+x^5+x^4+1) \oplus (x^6+x^4+x+1) \qquad (3)$$

Формула 3 тотожна перетворенню, записаному в формулі 1, проте переводить її до вигляду алгебри кінечних полів, в якій, в результаті і буде проводитися алгебраїчний аналіз шифру. При цьому, враховуючи те, що операція SubBytes() виконується першою в кожному із раундів шифрування, то не будемо забувати, що байт b' – це байт b, отриманий як результат попереднього раунду шифрування, розкладений над полем $Gf(2^8)$ із примітивним членом $x = x + 1$ та правилом заміни $x^8 = x^4 + x^3 + x + 1$.

Розкриття дужок між двома раундами шифру при цьому стає можливим після введення до рівняння усіченого диференціалу, що зображував би різницю між додаванням по модулю 2 та простим алгебраїчним додаванням.

$$(a \oplus b)' = (a + b - dif)' = \frac{a' \otimes b'}{dif'},$$
$$\text{де } dif = (a + b) - (a \oplus b) \qquad (4)$$

Введення цієї операції (4) може бути надефективним, адже якщо вірогідність розкриття дужок для кожного із байтів становить $\frac{1}{256}$, то вірогідність появи, скажімо 0-го диференціалу буде дорівнювати цілих 11.03668%. А знайшовши такий диференціал для кожного із раундів, можна буде знайти і ключ.

ВИСНОВКИ

Поєднавши, таким чином, пораундові рівняння, отримаємо систему рівнянь в алгебрі полів Галуа. Можливість її вирішення, або ж її лінеаризації, покаже актуальність атак на AES-подібні шифри за допомогою атак, основаних на алгебраїчних особливостях шифрів.

ЛІТЕРАТУРА

1. Bruce Schneier (2009-07-30). "Another New AES Attack". Schneier on Security, A blog covering security and security technology. Retrieved 2010-03-11.
2. Alex Biryukov; Orr Dunkelman; Nathan Keller; Dmitry Khovratovich; Adi Shamir (2009-08-19). "Key Recovery Attacks of Practical Complexity on AES Variants With Up To 10 Rounds". Retrieved 2010-03-11.
3. Alex Biryukov and Dmitry Khovratovich. Related-key Cryptanalysis of the Full AES-192 and AES-256 / Biryukov A., Khovratovich D. // University of Luxemburg.
4. Joan Daemen and Vincent Rijmen. The Design of Rijndael. AES — the Advanced Encryption Standard. Springer, 2002.

Отримано 27.06.2017

УДК 004.4

СИСТЕМИ ЗБОРУ ДАНИХ З ВЕБ-РЕСУРСІВ

Г. В. Тищенко, В. А. Святний, Ю. А. Бишенко
ДВНЗ «ДонНТУ», м. Покровськ

Проведено аналіз зростання обсягу даних в мережі Інтернет, розглянуто існуючі системи збору даних, показано, що їх архітектура потребує доробки. Розроблено архітектуру системи збору даних, яка допоможе позбавитися існуючих недоліків з використанням фреймворку Scrapy, бази даних PostgreSQL та платформи RabbitMQ.

Ключові слова: система збору даних, фреймворк Scrapy, база даних PostgreSQL, платформа RabbitMQ

У попередні роки відзначається експоненціальне зростання обсягу даних, також прогнозується продовження зростання в майбутньому. Тому задача написання системи програмного забезпечення для збору даних, яка ефективно накопичує дані високої якості, дуже важлива.

Кожна компанія зацікавлена в інформації про ринок, на якому вона працює. Для розвитку потрібні дані, які допомагають генерувати ці знання. Вони надають незаперечну перевагу над конкурентами, допомагаючи знаходити потенційних клієнтів, аналізувати реакцію споживачів на нові продукти, розміщувати більш ефективні стратегії маркетингу, а також розробляти більш якісні продукти.

Збір цієї інформації - складне завдання, коли основний підхід полягає в опитуванні людей. Однак, завдяки сучасним системам зв'язку, а особливо, інтернет-порталам, все більше даних надається у вільному доступі. У попередні роки відзначається експоненціальне зростання обсягу даних, також прогнозується продовження зростання в майбутньому. Основною перевагою збору є те, що дані, як правило, можуть бути отримані автоматично, за допомогою комп'ютерів.

Процес вилучення структурованої корисної інформації з сайту називається парсингом (parsing), а інструменти для реалізації даного процесу - парсерами (parsers).

Як правило, сайти розробляються з урахуванням того, що зчитувати інформацію з їх сторінок буде людина. Але формат представлених даних, зрозумілий людині, найчастіше не настільки зрозумілий програмним засобам. Крім того, структура представлених даних змінюється від сайту до сайту, тому не існує універсального засобу для вилучення інформації з них. Однак, існують готові

рішення, що дозволяють отримувати інформацію з сайту після попередньої конфігурації. Ці рішення часто коштують дорого і не мають тієї гнучкості, яку можуть дати рішення, розроблені під конкретний сайт.

Можна виділити наступні методи збору інформації з сайтів:

а) Ручний - в даному випадку роль парсеру виконує людина.

б) Гібридний - користувач як і раніше виконує основні дії для отримання інформації, але може використовувати допоміжні програмні засоби для автоматизації збору.

в) Автоматичний - отримання та структурування інформації виконується автоматично.

Для того, щоб один раз отримати значення декількох полів з 10-15 сторінок, писати окремий парсер недоцільно, однак в разі великого числа сторінок, полів або високої періодичності збору інформації, цей процес потребує автоматизації [1]. Тому завдання написання системи програмного забезпечення для збору даних, яка накопичує дані високої якості, дуже важлива.

Web Mining - це процес отримання даних з веб-ресурсів. Так як веб-джерела, як правило, не є текстовими даними, то і підходи до процесу вилучення даних відрізняються в цьому випадку. В першу чергу необхідно пам'ятати, що інформація в інтернеті зберігається у вигляді спеціальної мови розмітки HTML (RSS, Atom та інші), веб-сторінки можуть мати додаткову метаінформацію, а також інформацію про структуру (семантику) документа [6].

Підходи до вилучення даних:

а) аналіз DOM дерева;

б) використання XPath;

в) використання регулярних виразів;

г) візуальний підхід.

Перший підхід базується на аналізі DOM дерева. DOM (Document Object Model - «об'єктна модель документа») - це не залежний від платформи та мови програмний інтерфейс, що дозволяє програмам та скриптам отримати доступ до вмісту HTML, XHTML і XML-документів, а також змінювати структуру та оформлення таких документів.

На рис. 1 зображено модель DOM - дерева.

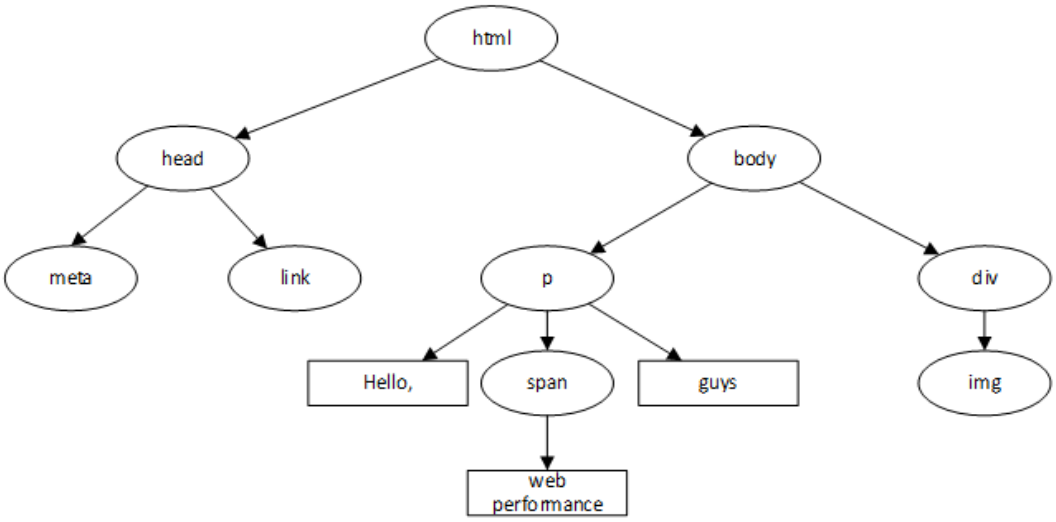


Рисунок 1 - Модель DOM – дерева

Використовуючи цей підхід, дані можна отримати безпосередньо за ідентифікатором, ім'ям або іншим атрибутом елемента дерева (таким елементом може служити параграф, таблиця, блок і т.п.).

Наступним еволюційним етапом аналізу DOM-дерева є використання XPath - це мова запитів до елементів xml- або xhtml-документа. Суть даного підходу в тому, щоб за допомогою деякого простого синтаксису описувати шлях до елементу без необхідності поступового руху вниз DOM- деревом.

Регулярні вирази – це система обробки тексту, заснована на спеціальній системі запису шаблонів для пошуку.

Візуальний підхід в даний момент знаходиться на початковій стадії розвитку. Суть підходу полягає в тому, щоб користувач міг без використання програмної мови або API «налаштувати» систему для отримання потрібних даних будь-якої складності та вкладеності [8].

У Web Mining можна виділити наступні етапи:

а) вхідний етап (input stage) - отримання "сирих" даних з джерел;

б) етап обробки (preprocessing stage) - дані подаються у формі, необхідній для успішної побудови тієї чи іншої моделі;

в) етап моделювання (pattern discovery stage);

г) етап аналізу моделі (pattern analysis stage) - інтерпретація отриманих результатів.

Існує багато систем для вилучення даних з інтернет-джерел. Найпопулярніші парсери - Datacol, Content Downloader, X-Parser Light, FDE Grabber, WP Uniparser.

Всі вони мають такі недоліки:

а) працюють повільно;

б) багато разів програма звертається в базу даних або в файл;

в) багато часу займають налаштування;
г) збирається менше 95% даних з порталів;
ґ) іноді на виході отримуємо помилковий контент;
д) виникають помилки при роботі з великими обсягами даних.

В результаті аналізу цих недоліків було розроблено архітектуру, зображену на рис.2.

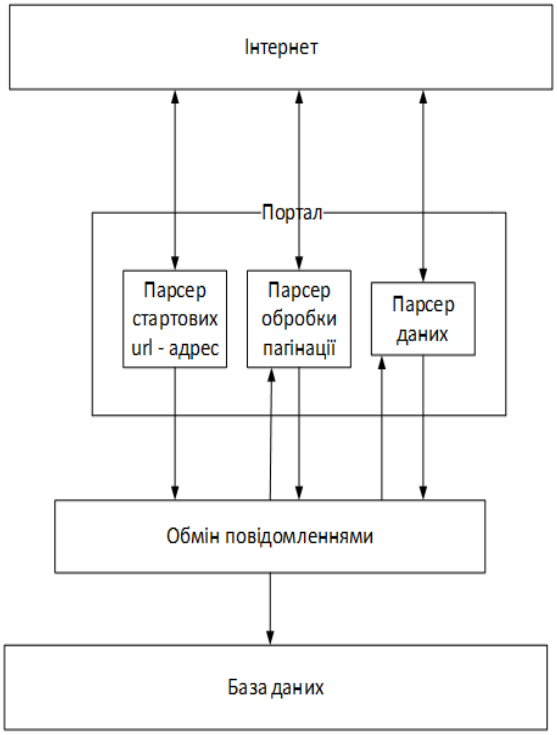


Рисунок 2 - Архітектура системи збору даних з веб-ресурсів

У результаті дослідження було виявлено, що обсяг даних в мережі Інтернет зростає за експоненціальною залежністю, тому необхідно розробляти та оптимізувати системи збору даних з веб-ресурсів. Розглянувши існуючі системи збору даних, можна зробити висновок про те, що їх архітектура потребує доробки. Тому, було розроблено архітектуру системи збору даних, яка допоможе позбавитися існуючих недоліків. Для реалізації такої системи було використано фреймворк Scrapy, база даних – PostgreSQL та платформа RabbitMQ.

ЛІТЕРАТУРА

1. Web parsing: задачи, проблемы, инструменты [Електронний ресурс]: «Inostudio». – Режим доступу: https://inostudio.com/ru/article/web-parsing.html – Дата доступу: березень 2017. – Загол. з екрану.

2. Data Mining [Електронний ресурс]: «UserGroup». – Режим доступу: http://msugvnua000.web710.discountasp.net/Posts/Details/3316 – Дата доступу: березень 2017. – Загол. з екрану.
3. An introduction to big data [Електронний ресурс]: «Opensource». – Режим доступу: https://opensource.com/resources/big-data – Дата доступу: березень 2017. – Загол. з екрану.
4. Big Data and Data Mining [Електронний ресурс]: «Habrahabr». – Режим доступу: https://habrahabr.ru/post/267827/– Дата доступу: березень 2017. – Загол. з екрану.
5. Data Mining [Електронний ресурс]: «Habrahabr». – Режим доступу: https://habrahabr.ru/post/95209/– Дата доступу: березень 2017. – Загол. з екрану.
6. Document Object Model [Електронний ресурс]: «W3C». – Режим доступу: https://www.w3.org/DOM/ – Дата доступу: березень 2017. – Загол. з екрану.
7. Подходы к извлечению данных с веб-ресурсов [Електронний ресурс]: «Habrahabr». – Режим доступу: https://habrahabr.ru/post/99918/ – Дата доступу: березень 2017. – Загол. з екрану.
8. Подходы к извлечению данных с веб-ресурсов [Електронний ресурс]: «Habrahabr». – Режим доступу: https://habrahabr.ru/company/dataart/blog/262817/ – Дата доступу: березень 2017. – Загол. з екрану.
9. Web Mining: интелектуальный анализ данных в сети Internet [Електронний ресурс]: «Google». – Режим доступу: https://sites.google.com/site/upravlenieznaniami/tehnologii-upravlenia-znaniami/text-mining-web-mining/web-mining – Дата доступу: березень 2017. – Загол. з екрану.
10. Datacol – универсальный парсер с визуализацией сбора данных [Електронний ресурс]: «Profithunter». – Режим доступу: https://www.profithunter.ru/obzory/datacol-universalnyiy-parser-s-vizualizatsiey-sbora-dannyih/– Дата доступу: березень 2017. – Загол. з екрану.
11. Datacol – парсер для сбора информации с сайтов [Електронний ресурс]: «Vlada-rykova». – Режим доступу: https://vlada-rykova.com/parser-sajtov/– Дата доступу: березень 2017. – Загол. з екрану.
12. Универсальный парсер контента – программа Content Downloader [Електронний ресурс]: «Seobook». – Режим доступу: http://seobook.info/universalnyy-parser-kontenta/ – Дата доступу: березень 2017. – Загол. з екрану.
13. Перелинковка сайта [Електронний ресурс]: «Pr-cy». – Режим доступу: http://pr-cy.ru/lib/saytostroenie/Perelinkovka-sayta-Kak-eto-sdelat – Дата доступу: березень 2017. – Загол. з екрану.

Отримано 20.06.2017

Секція 2.

High Performance Computing (HPC) and Parallel Simulation Technology (ParSimTech)

BRINGING TOGETHER HIGH PERFORMANCE COMPUTING AND DATA ANALYTICS

Michael Resch

High Performance Computing is facing limitations by the nearing end of Moore's law. Still we can see an increase in performance but the rate of acceleration decreases continuously. With this in mind science turns towards another problem which has emerged over the last years. Ever faster supercomputers have created ever larger amounts of data. This phenomenon used to be considered to be part of what is called Big Data. However, this cannot be separated from high performance computing. In this talk it will be explained how the analysis of data and the usage of high performance computing can be brought together to create new solutions both in science and in industry.

Received 29.06.2016

УДК 004.92

NONLINEAR MODEL REDUCTION OF POPULATION BALANCE EQUATIONS BY POD-DEIM BASIS CONSTRUCTION

Michael Mangold[1], Dmytro Khlopov[1], Lihong Feng[2]
[1]Technische Hochschule Bingen, Berlinstr. 109, 55411 Bingen, Germany
[2]Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany

A rather general method for the nonlinear model reduction of differential algebraic system is presented. The method can be automatized to a large degree and has been integrated in a flowsheet simulator. It is applied to the model of a crystallization process consisting of partial differential equations for a fluid phase and a disperse phase. A good agreement between the full and the reduced model is achieved.
Key words: nonlinear model reduction; proper orthogonal decomposition; empirical interpolation; population balance; crystallization.

1. INTRODUCTION

Particulate systems consisting of a continuous fluid phase and a disperse particle phase play an important role in the chemical and pharmaceutical industry, because the majority of products take on a crystalline form at some point of the production process. Physical modeling of particulate systems has to account for a continuous fluid phase with complex fluid dynamics, and for a disperse particulate phase containing crystals or other particles of different size, shape, or composition. Mass, momentum, and energy balances of the continuous phase lead to partial differential equations in space with up to three space coordinates. The model of the disperse phase may describe the interactions of single particles with each other and with the fluid. This results in highly detailed, computationally extremely expensive CFD-DEM simulations (Zhou et al., 2010). An alternative approach, applicable to disperse phases with a very high number of particles, is to capture the behavior of the complete particle population instead of single particles. This results in population balance equations, whose independent variables are time, space coordinates, and, in addition, property coordinates like e.g. the particle size (Ramkrishna, 2000). Nevertheless, the result of the modeling efforts is a coupled system of nonlinear partial differential equations, whose numerical solution is challenging and computationally expensive. For purposes of controller design and process optimization, simpler model formulations are required. This motivates the search for model reduction techniques that distill a low order model out of the detailed reference model. In this work, Proper Orthogonal Decomposition (POD) in combination with the Discrete Empirical Interpolation Method (DEIM) is used to construct the reduced model.

2. MODEL REDUCTION TECHNIQUE

The approach is illustrated by the model of a continuous fluidized bed crystallizer from (Mangold et al., 2015; Mangold et al., 2017). The key equation of the model is a population balance for the number size density $n(x, L, t)$, which has the following structure:

$$\frac{\partial n}{\partial t} = -\frac{\partial}{\partial x}(v_P n) + D\frac{\partial^2 n}{\partial x^2} - G\frac{\partial n}{\partial L}.$$

The variable x is the geometrical (length) coordinate of the crystallizer, L is the size coordinate of the crystals, t is the time; v_p is the particle velocity in space, which depends in a nonlinear way on L and $n(x, L, t)$; D is a dispersion coefficient; G is the growth rate of the crystals.

A semi-discretization of the model using the Method of Lines turns the model equation into a set of differential algebraic equations of the form

$$E\frac{d\boldsymbol{x}}{dt} = A\,\boldsymbol{x} + \boldsymbol{f}(\boldsymbol{x}),$$

where $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{N}}$ is the discretized state vector, A, E are constant matrices, and $\boldsymbol{f}: \mathbb{R}^N \to \mathbb{R}^N$ comprises the nonlinearities of the system.

The idea of the model reduction technique is to first perform test simulations with the reference model, and to collect the dynamic simulation results in a so-called snapshot matrix $\boldsymbol{X} \in \mathbb{R}^{M\times N}$, M being the number of available time steps. Using Singular Value Decomposition, $\boldsymbol{X}$ can be written as

$$\boldsymbol{X} = \widetilde{\boldsymbol{U}}\,\boldsymbol{\Sigma}\,\widetilde{\boldsymbol{V}},\ \ \widetilde{\boldsymbol{U}} \in \mathbb{R}^{M\times N}, \widetilde{\boldsymbol{V}} \in \mathbb{R}^{N\times N}.$$

The first r columns of $\widetilde{\boldsymbol{V}}$, denoted as $\boldsymbol{V} \in \mathbb{R}^{N\times r}$, are used as reduced basis vectors of the full state vector $\boldsymbol{x}$, i.e.

$$\boldsymbol{x} \approx \boldsymbol{V}\boldsymbol{z},$$

where $\boldsymbol{z} \in \mathbb{R}^r$ is the reduced state vector, whose dimension r should be considerably smaller than N. Inserting the approximation into the discretized model equations gives a first formulation of the reduced model:

$$\boldsymbol{V}^T E\,\boldsymbol{V}\frac{d\boldsymbol{z}}{dt} = \boldsymbol{V}^T A\,\boldsymbol{V}\,\boldsymbol{z} + \boldsymbol{V}^T\boldsymbol{f}(\,\boldsymbol{V}\,\boldsymbol{z})$$

Evaluating the nonlinear term of the reduced model in the above form would be extremely inefficient, because computing the full size vector of nonlinearities is still unavoidable. A drastic speed-up of the reduced model is achieved by approximating the nonlinearity by a reduced basis, as well:

$$\boldsymbol{f} \approx \boldsymbol{U}_{\boldsymbol{f}}\boldsymbol{c},$$

where $\boldsymbol{U}_{\boldsymbol{f}}$ is a POD basis of $\boldsymbol{f}$. This technique is called Discrete Empirical Interpolation Method (Chaturantabut and Sorensen, 2010). For further details on the model reduction method, the reader is referred to (Feng et al., 2017)

3. AUTOMATIC MODEL REDUCTION

The big advantage of the POD/DEIM approach is that it can be automatized to a large extent. A software tool performing the model reduction task with minimum input from the user has been developed by Khlopov and Mangold (2017). The tool is a part of the process modeling system ProMoT. It is applicable to any differential algebraic system implemented in the model definition language MDL. Using an interface to the computer algebra system Maxima, the model reduction tool analyses the model code and generates the reduced model equations in symbolic form. Numerical operations necessary to generate the basis functions of the reduced model are performed in the simulation tool Diana and in Octave. Due to the modular structure of the

software code, these numerical tools could be replaced easily by other equation solvers and numerical libraries.

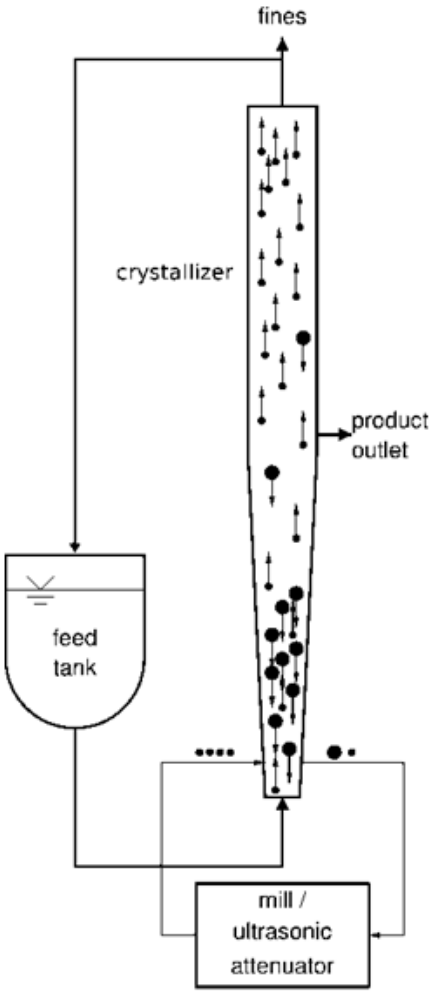
4. APPLICATION EXAMPLE

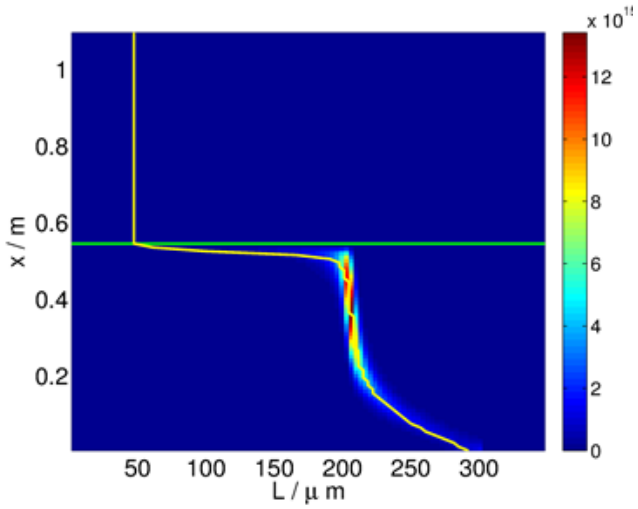
As an application example, a novel process for the separation of enantiomers has been studied. The key element of the process is a tubular crystallizer as depicted in Figure 1. A fluid containing a mixture of two enantiomers in the liquid phase flows through the crystallizer from bottom to top. The crystallizer contains seeding crystals of one enantiomer. The operation conditions are chosen such that the crystals of that enantiomer grow selectively while the other enantiomer remains in the liquid phase. Small crystals are moving to the top of the crystallizer with the fluid flow; large crystals sink to the bottom. They are withdrawn from the crystallizer, crushed to small fragments in a mill, and recycled to the process as new seeds.

Simulation results obtained with a first principle model of the process are also shown in Figure 1. After discretization of space and property coordinates, the first principle model consists of 9800 ordinary differential equations (ODEs). The model reduction tool presented in (Khlopov and Mangold, 2017) generates a reduced model of 45 ODEs and 66 algebraic equations. A comparison between the reference model and the reduced model in Figure 2 shows a good agreement between both of them.

5. CONCLUSIONS

Model reduction by proper orthogonal decomposition is an attractive approach for obtaining low order models from detailed reference models of chemical processes. In this example, it has been applied to a crystallization system described by population balances, but due to the general nature of the approach, many other applications are possible. The method is very appealing because it can be included in a software tool that performs the model reduction automatically. Adaptive basis construction and error estimation, as described in (Feng et al., 2017) provide reliable reduced models of low order suitable for controller design or optimization problems.

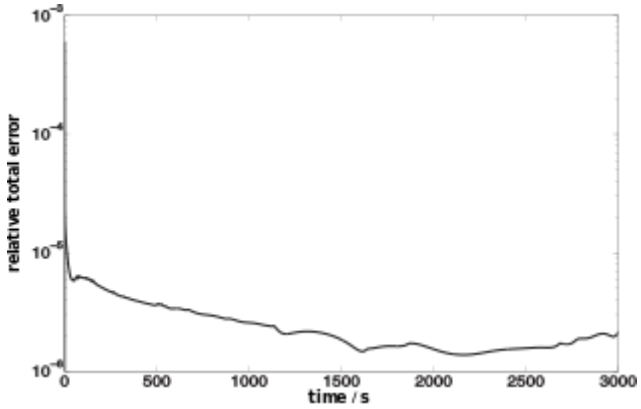




¶

Figure·1: left-hand side: scheme of crystallization process for separation of enantiomers; diagram above: number density function of crystals under steady state conditions; x=length coordinate of crystallizer; L=size coordinate of crystals; the green line marks the position of the product outlet; the yellow line indicates the size of crystals in equilibrium with the fluid flow.¤

Figure 2: Relative error of the reduced POD / DEIM model.



BIBLIOGRAPHY:

Chaturantabut, S., Sorensen, D. (2010): Nonlinear model reduction via discrete empirical interpolation. SIAM Journal on Scientific Computing 32:2737-2764.

Feng, L., Mangold, M., Benner, P. (2017): Adaptive POD-DEIM basis construction and its application to a nonlinear population balance system. AIChE Journal, in press.

Khlopov, D., Mangold, M. (2017): Automatic model reduction of differential algebraic systems by proper orthogonal decomposition. Computers & Chemical Engineering 97: 104-113.

Mangold, M., Feng, L., Khlopov, D., Palis, S., Benner, P., Binev, D., Seidel-Morgenstern, A. (2015). Nonlinear model reduction of a continuous fluidized bed crystallizer. Journal of Computational and Applied Mathematics 289:253-266.

Mangold, M. Khlopov, D., Temmel, E., Lorenz, H., Seidel-Morgenstern, A. (2017): Modelling geometrical and fluid-dynamics aspects of continuous fluidized bed crystallizer for separation of enantiomers. Chemical Engineering Science 160: 281-290.

Ramkrishna, D. (2000): Population Balances: Theory and Applications to Particulate Systems in Engineering. Academic Press, New York.

Zhou, Z., Kuang, S., Chu, K., Yu, A. (2010): Discrete particle simulation of particle-fluid flow: model formulations and their applicability. Journal of Fluid Mechanics 661:482-510.

Received 29.06.2017

УДК 004.75

ВИКОРИСТАННЯ СПЕЦІАЛІЗОВАНОГО ПРИКЛАДНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ RAINBOW У НАУКОВО-ДОСЛІДНИХ ГРІД-ПРОЕКТАХ

С.Д. Погорілий, д-р техн. наук, проф.,
Ю.В. Бойко, к.ф.-м.н., доцент, А.О. Сальніков, к.т.н., асистент,
Є.А. Слюсар к.т.н., асистент, О.Ф. Борецький, асистент
Київський національний університет ім. Тараса Шевченка

Розглянуто особливості використання спеціалізованого прикладного програмного забезпечення у науково-дослідних грід-проектах. Описано особливості використання комплексу програмного забезпечення Rainbow для виконання прикладного програмного забезпечення у віртуальних машинах запущених як грід завдання. Запропоновано методи формування, конфігурування образів віртуальних та встановлення потрібного прикладного програмного забезпечення перед запуском віртуальної машини

Ключові слова: грід, віртуальна машина, Rainbow, прикладне програмне забезпечення, Windows, GNU/Linux, Docker

ВСТУП

Для вирішення сучасних ресурсоємних науково-дослідних задач в різних галузях науки і техніки в Україні побудовано академічний грід-сегмент, розвинутий до національного масштабу який формує понад 30 кластерів у провідних науково-дослідних інститутах та університетах [1]. У порівнянні з іншими країнами світу характерною особливістю українського національного грід є використання грід-технологій в різних галузях науки - фізиці, хімії, біології, астрономії, медицині, тощо, тоді як більшість Європейських грід-інфраструктур зосереджені лише на задачах фізики високих енергій великого різних експериментів адронного коллайдера [2].

Переважна більшість проектів використовує власне спеціалізоване прикладне програмне забезпечення яке для певних потреб потребує графічного інтерфейсу розроблене виключно під певне сімейство операційних систем(ОС) та потребує наявності інтерактивного доступу до нього.

У роботі [5] було запропоновано використати грід-інфраструктуру як платформу для віртуалізації. Даний підхід передбачає у звичайних грід-завданнях запуск віртуальних машин з апаратним прискоренням та контейнерних машин, що в свою чергу дозволяє

створити ізольоване, контрольоване середовище для запуску спеціалізованого прикладного програмного забезпечення. Однак потребує методів та засобів які дозволять динамічно налаштовувати та модифікувати образи віртуальних машин перед їх використанням.

ОСОБЛИВОСТІ ВИКОРИСТАННЯ ПРИКЛАДНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У НАКОВО-ДОСЛІДНИХ ГРІД-ПРОЕКТАХ

Одним з найбільших грід-проектів який динамічно розвивається в Україні в області медицини є проект «Медгрід» [3]. На першому етапі розвитку проекту було вирішено завдання накопичення великих об'ємів медичних даних в грід-інфраструктурі для їх надійного зберігання. Дані, що потрапляють до грід-інфраструктури через спеціалізований веб-портал автоматично реплікуються поміж різними сховищами даних за допомогою сервісу РАПТОР [4].

На актуальному етапі розвитку проекту виконується аналіз популяційні дослідження на основі кардіологічних та спірометричних аналізів також проводиться в грід-інфраструктурі. Якісний аналіз медичних досліджень потребує інтерактивного аналізу за допомогою спеціалізованого програмного забезпечення для ОС Windows в інтерактивному режимі та з візуалізацією. Для вирішення даного завдання було розроблено комплекс програмного забезпечення Rainbow, що реалізує парадигму платформа-як-сервіс в грід, та дозволяє забезпечує інтерактивний доступ до віддаленого робочого столу віртуальної машини, що працюють як грід-задачі [5].

Наступним кроком інтерактивного аналізу медичних даних є проведення відкладених медичних консультацій для первинної ланки надання медичної допомоги рис.1. Розробка пілотного сервісу, що реалізує платформу для телемедицини з використанням ресурсів УНГ та Rainbow є актуальною задачею, що активно розв'язується проектом [6].

З робочого місця лікаря першої ланки надання медичної допомоги здійснюється надсилання деперсоналізованих медичних досліджень до сервісу накопичення медичних даних у грід-інфраструктурі, який забезпечує їх надійне зберігання.

Для надання консультації розробляється сервіс який дозволить, у разі потреби надання відкладених телемедичних консультацій, створювати віртуальну машина, запущену як грід-задача, з даними для аналізу та візуалізації, доступ до якої лікарю-діагносту надається засобами віддаленого робочого столу.

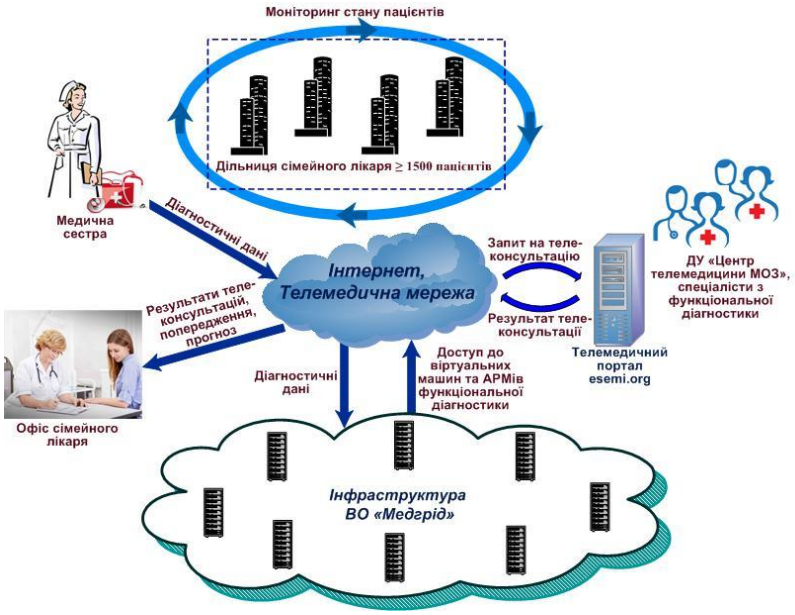


Рисунок 1 - Архітектура сервісу телемедичних консультацій

Ефективне проведення відкладених консультацій потребує застосування різних наборів пакетів програмного забезпечення (ППЗ) в одній віртуальній машині для різних типів діагностичних даних. На даний момент для кожного пакету діагностичного програмного забезпечення, що використовується в проекті «Медгрід», створено окремі образи віртуальних машин.

Даний підхід передбачає окремі образи віртуальних машин для кожного набору прикладного програмного забезпечення або відповідних комбінацій, що потребує великого об'єму для збереження всіх образів віртуальних машин і ускладнює їх обслуговування.

Іншим українським грід-проектом який динамічно розвивається є проект віртуальної лабораторії MolDynGrid який підтримується більшістю обчислювальних кластерів українського національного грід. Лабораторія надає науковцям зручний інструментарій для проведення розрахунків молекулярної динаміки біологічних макромолекул (білків, нуклеїнових кислот та їхніх комплексів) у водно-іонному оточенні з наближенням до фізіологічних умов [7-8]. Із використанням сервісів MolDynGrid на сьогодні проводяться дослідження молекулярної динаміки аміноацил-тРНК синтетаз та їх мутантних форм, що є ключовими ферментами на дорибосомному етапі біосинтезу протеїнів. За результатами досліджень було підтверджує гіпотезу Шиммеля про відсутність цитокінової активності (проангіогенні властивості, що полягають у стимуляції розвитоку кровоносних судин) за рахунок екранування цитокіного ELR-мотиву в повнорозмірній структурі тирозил-тРНК синтетази людини [9-10].

Виявлено локальні конформаційні зміни із формуванням β-структурних елементів в CP1-вставці згортки Россмана для мутантних форм G41R і del153-156VKQV тирозил-тРНК синтетази людини, що асоційовані з нейрологічним захворюванням Шарко-Марі-Туса (DI-CMTC) [11]. Запропоновано нову роль неструктурованої петлі CP1 вставки при взаємодії з низькомолекулярними субстратами [12].

При проведенні комп'ютерних моделювань науковцями використовується спеціалізоване програмне забезпечення GROMACS, NAMD та інші у різних конфігураціях і різних версій з використанням великої інфраструктури лабораторії рис. 2 та ряду існуючих засобів на кшталт утиліти Moldynsub_CLI.

Підтримка, обслуговування та оновлення цього прикладного програмного забезпечення на всіх обчислювальних кластерах грід-інфраструктури які підтримують віртуальну лабораторію MolDynGrid створює значне адміністративне навантаження на технічних спеціалістів проекту.

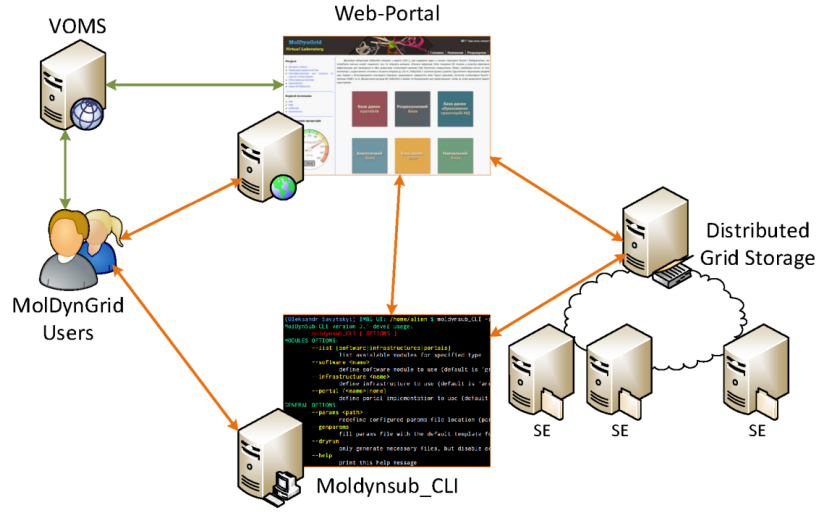


Рисунок 2 - Сервіси віртуальної лабораторії MolDynGrid

Для вирішення даної проблеми у попередніх роботах було запропоновано та використано виконання спеціалізованого прикладного програмного забезпечення у контейнерній віртуальній машині запущеній як грід-завдання. Даний підхід дозволив запустити потрібне прикладне програмне забезпечення у ізольованому, незалежному від архітектури обчислювального кластера, керованому середовищі контейнерної віртуальної машини без накладних витрат на емуляцію апаратних компонент комп'ютера.

ОСОБЛИВОСТІ ВИКОРИСТАННЯ КОМПЛЕКСУ RAINBOW ДЛЯ ВИКОНАННЯ ПРИКЛАДНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Rainbow [5] – комплекс програмного забезпечення який реалізовує запуск віртуальних машин як грід-завдань. Складається з елементів (рис. 3):

- Обчислювального грід-вузла(налаштування середовища виконання);
- Обчислювального вузла кластера;
- Мережевого шлюзу.

Елемент обчислювального грід-вузла виконує передачу параметрів з файлу опису грід-завдання та завантаження вхідних даних.

Елемент обчислювального вузла кластера виконує всі потрібні завдання попередньої підготовки до запуску віртуальної машини та керує її життєвим циклом.

Елемент Мережевого шлюзу забезпечує мережеву конфігурацію для віртуальної машини та інтерактивний доступ до неї із зовнішніх мереж.

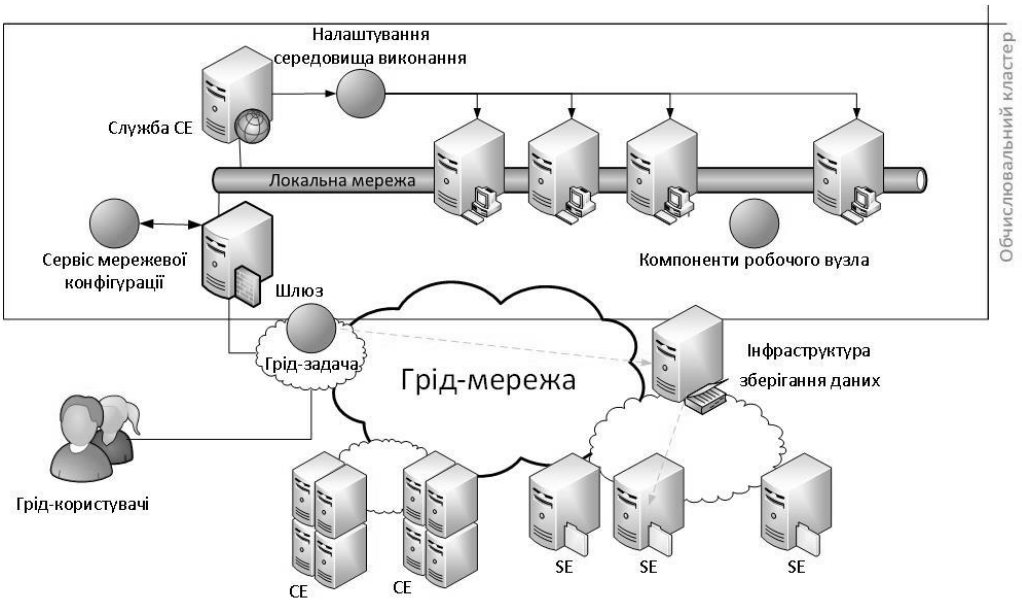


Рисунок 3 - Компоненти Rainbow

При запуску віртуальної машини віртуальної машини як грід-завдання з обчислювального елементу грід виконується передача параметрів віртуальної машини та інтерактивного доступу, а також образу віртуальної машини і вхідних файлів у разі потреби безпосередньо до обчислювального вузла кластера. Образи віртуальних машин та вхідні файли зберігаються у розподіленій інфраструктурі зберігання

даних грід. Образ віртуальної машини це попередньо встановлена ОС бібліотеки та потрібне програмне забезпечення[13].

На обчислювальному вузлі кластера виконується запуск віртуальної машини із заданими параметрами та передача вхідних файлів до віртуальної машини.

Після запуску віртуальної машини вона надсилає запит на отримання мережевої конфігурації по протоколу DHCP. Після видачі IP-адреси виконується налаштування мережевого екрану для перенаправлення портів для протоколів SSH або RDP що забезпечує інтерактивний доступ до запущеної віртуальної машини.

ЛІТЕРАТУРА

1. Базовий координаційний центр УНГ [Електронний ресурс] Режим доступу http://ung.in.ua/ua/resourses/.
2. Український академічний Грід: досвід створення й перші результати експлуатації / Ю.В. Бойко, М.Г. Зинов'єв, О.О. Судаков, С.Я. Свістунов // Математичні машини і системи. – 2008. – Випуск 1. – С. 67–84
3. Проект «Медгрід» [Електронний ресурс] Режим доступу http://medgrid.immsp.kiev.ua/
4. Слюсар, Е. А., & Волжева, М. В. Использование грид-инфраструктуры как распределенного отказоустойчивого хранилища данных для веб-сервисов.
5. Борецкий А.Ф., Вишневский В. В., Сальников А. А. «Платформа как сервис» в ГРИД для интерактивного анализа медицинских данных. Математические машины и системы, 2015, 1.
6. Вишневський, В. В., Осташко, В. Г., Динник, О. Б., & Мостовий, С. Є. використання ресурсів ВО «Медгрід» для відкладених телеконсультацій лікарем з функціональної діагностики.
7. Salnikov A.O., Sliusar I.A., Sudakov O.O., Savytskyi OV, Kornelyuk O.I. MolDynGrid Virtual Laboratory as a part of Ukrainian Academic Grid infrastructure// Proceedings of IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications. - 2009. - P. 237-240.
8. Salnikov A., Sliusar I., Sudakov O., Savytskyi O. and Kornelyuk A. / Virtual laboratory MolDynGrid as a part of scientific infrastructure for biomolecular simulations. // International Journal of Computing. — 2010. — V. 9, N 4. — P. 294-300.
9. Yang X.-L., Skene R.J., McRee D.E. and Schimmel P. / Crystal structure of a human aminoacyl-tRNA synthetase cytokine // Proc. Natl. Acad. Sci. U. S. A. — 2002. — V. 99, N 24. — P. 15369-15374.
10. Savytskyi O. V. and Kornelyuk A. I. / Computational modeling of molecular dynamics of G41R mutant form of human tyrosyl-tRNA synthetase, associated with Charcot-Marie-Tooth neuropathy // Ukr Biochem J. — 2015. — V. 87, N 6. — P. 142-53.
11. High Throughput Computing [Електронний ресурс] – Режим доступу: https://wiki.egi.eu/wiki/Glossary_V1#High_Throughput_Computing
12. OSG Usage Graphs [Електронний ресурс] – Режим доступу: http://display.grid.iu.edu/

13. Boretskyi, O., Salnikov, A., Sliusar, I., Sudakov, O., & Boyko, Y. (2015, September). Rainbow framework: Running virtual machines on demand as a grid jobs. In Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2015 IEEE 8th International Conference on (Vol. 2, pp. 972-976). IEEE.
14. Honeycutt, J. (2012). Introducing Windows 8. Microsoft.
15. Wise Solutions Installer [Електронний ресурс] Режим доступу http://www.installaware.com/wise-comparison.htm
16. Nullsoft Scriptable Install System [Електронний ресурс] Режим доступу http://nsis.sourceforge.net/Main_Page.
17. InstallShield [Електронний ресурс] Режим доступу http://www.flexerasoftware.com/producer/products/software-installation/installshield-software-installer/
18. Inno Setup [Електронний ресурс] Режим доступу http://www.jrsoftware.org/isinfo.php
19. Microsoft Installer [Електронний ресурс] Режим доступу https://msdn.microsoft.com/en-us/library/aa372463(v=vs.85).aspx
20. Windows Installer Xml (WiX) toolset [Електронний ресурс] Режим доступу http://wixtoolset.org/
21. Unattended setup [Електронний ресурс] Режим доступу https://technet.microsoft.com/en-us/library/cc785644(v=ws.10).aspx.

Отримано 26.06.2017

Секція 3.
Комп'ютерна графіка та візуалізація в ігрових додатках

УДК 004.9

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ПАРАЛЕЛЬНОЇ РЕАЛІЗАЦІЇ АЛГОРИТМУ БІЛАТЕРАЛЬНОЇ ФІЛЬТРАЦІЇ ДЛЯ ВИРІШЕННЯ ЗАВДАНЬ РОЗПІЗНАВАННЯ ОБРАЗІВ НА ЗОБРАЖЕННЯХ ЗАСОБАМИ ТЕХНОЛОГІЇ CUDA

М.Ю. Драченко, С.А. Зорі

Донецький Національний Технічний Університет,
м. Покровськ, Україна

У роботі висвітлюється метод білатеральної фільтрації та можливості його паралельної реалізації для розпізнавання образів на зображеннях. Доводиться, що динамічне ядро фільтру більш вигідне, ніж статичне. Досліджуються можливості технології CUDA та проводиться аналіз можливості відображення білатерального фільтру та алгоритму регіонів на паралельну топологію.

Ключові слова: зображення, обробка, піксель, фільтрація, канал, алгоритм, потік, багатопоточність, CUDA, мережа, топологія.

ВСТУП. АКТУАЛЬНІСТЬ

Цифрова обробка зображень (ЦОЗ) відноситься до цифрової обробки сигналів, за умови що сигнал – зображення. ЦОЗ є окремою галуззю знань, що поширюється наразі достатньо швидко. ЦОЗ охвачує великий спектр алгоритмів і методів, що застосовуються в різних галузях людської діяльності. Одна з найважливіших можливостей ЦОЗ – покращення якості зображення. Можливість покращити зображення допомагає не тільки збільшити якість власних фотографій користувачів конкретних програмних продуктів, а ще й, наприклад, збільшити чіткість зображення електронного мікроскопу, допомогти в розробці певних ділянок Землі задля добутку корисних копалин через покращення

зображень зі супутників Землі, а також покращити зображення медичного характеру, що грає велику роль у сьогоденній медицині.

Зображення можна представити різними засобами, наприклад, це може бути певна програма, що буде виконувати певну послідовність дій (так звана векторна графіка), але в роботі буде запропонована робота із растровою графікою. Растрова графіка – це приклад зображення, що представлено, як матриця пікселів. Тобто можна послідовно обробити будь-яке растрове зображення певними алгоритмами фільтрів або аналізу певних характеристик зображення. Звісно, ці параметри інколи не є об'єктивними показниками конкретної проблеми, бо на числах певний показник може буди незадовільнім, а на практиці – людський зір адаптується до гамми кольорів. Очевидно, що людським зором, як органом відчуття людини, комп'ютери поки що не здатні заволодіти, тому виникла галузь комп'ютерних наук, що називається «Комп'ютерне бачення», або «Комп'ютерний зір».

Комбінація цифрової обробки зображень та комп'ютерного зору дозволяє класифікувати зображення та навіть розпізнавати певні об'єкти не тільки на зображеннях, а й на відео. У цій роботі буде запропоновано методи прискорення роботи алгоритмів завдяки можливостям багатопоточності та застосуванню графічних процесорів із допомогою технології CUDA. Технологія CUDA дозволяє створити певну мережу процесорів та відтворити на цю мережу певний алгоритм, щоб прискорити виконання алгоритму.

1. БІЛАТЕРАЛЬНА ФІЛЬТРАЦІЯ

Білатеральна фільтрація – це нелінійний вид фільтрації цифрового зображення, що вперше був опублікований в статті [1]. Цей вид фільтрації активно застосовується для позбавлення від шуму, а також для розпізнавання точкових цілей, що рухаються. [2][3] Вихідне значення яскравості відфільтрованого зображення у точці (i, j) можна обчислити за формулою (1):

$$Y(i,j) = \sum_{l=-N}^{N} \sum_{k=-M}^{M} H(i,j;l,k)X(l,k) \qquad (1)$$

У формулі H(i,j; l, k) – ядро білатерального фільтру, а X(i,j) – значення інтенсивності у точці (i,j) початкового зображення.

Ядро білатерального фільтру має наступний вид (2)[4]:

$$H(i,j,l,k) = \begin{cases} w^{-1}(i,j)e^{\left(-\frac{(l-i)^2+(k-j)^2}{2\sigma_d^2}\right)}e^{\left(-\frac{(X(l,k)-X(i,j))^2}{2\sigma_r^2}\right)}, (l,k) \in \Lambda_{i,j} \\ 0, (l,k) \notin \Lambda_{i,j} \end{cases} \quad (2)$$

де Λi,j – ковзне вікно фільтрації;

σ_d^2 – просторова дисперсія;

σ_r^2 – рангова дисперсія;

$w_{i,j}^{-1}$ – коефіцієнт нормування, що має наступний вигляд (3):

$$w(i,j) = \sum_{l=i-N}^{N} \sum_{k=j-M}^{M} e^{\left(-\frac{(l-i)^2+(k-j)^2}{2\sigma_d^2}\right)}e^{\left(-\frac{(X(l,k)-X(i,j))^2}{2\sigma_r^2}\right)} \quad (3)$$

Білатеральна фільтрація дуже добре працює з будь-яким видом шуму, але істотній недолік щодо результату є розмиття маленьких об'єктів на зображенні. Проте цей алгоритм не розмиває, а ще й зберігає, великі об'єкти у межах їх країв, що є великою перевагою над іншими алгоритмами.

2. ЗАСТОСУВАННЯ ТЕХНОЛОГІЇ CUDA ДЛЯ ОБРОБКИ ЗОБРАЖЕНЬ

Одне зображення високої роздільної здатності може містити в собі близько двох мільйонів пікселів. Багато алгоритмів обробки зображень вимагають десятки обчислень з плавучою точкою на піксель, що може привести до повільного виконання навіть для самих швидких процесорах. Повільна швидкість процесора є серйозною перешкодою для продуктивності. У CUDA, так як зображення звичайно двомірні, має сенс мати кожен блок сітки двовимірним. 32 × 16 - хороший розмір, оскільки він дозволяє кожному блоку запустити 512 потоків. Потім можна створити стільки блоків потоків, скільки необхідно для покриття всього зображення. Наприклад, для 1024 × 768 зображення, сітка потокових блоків становить 32 × 48, а кожний блок має розміри 32 × 16 потоків [6].

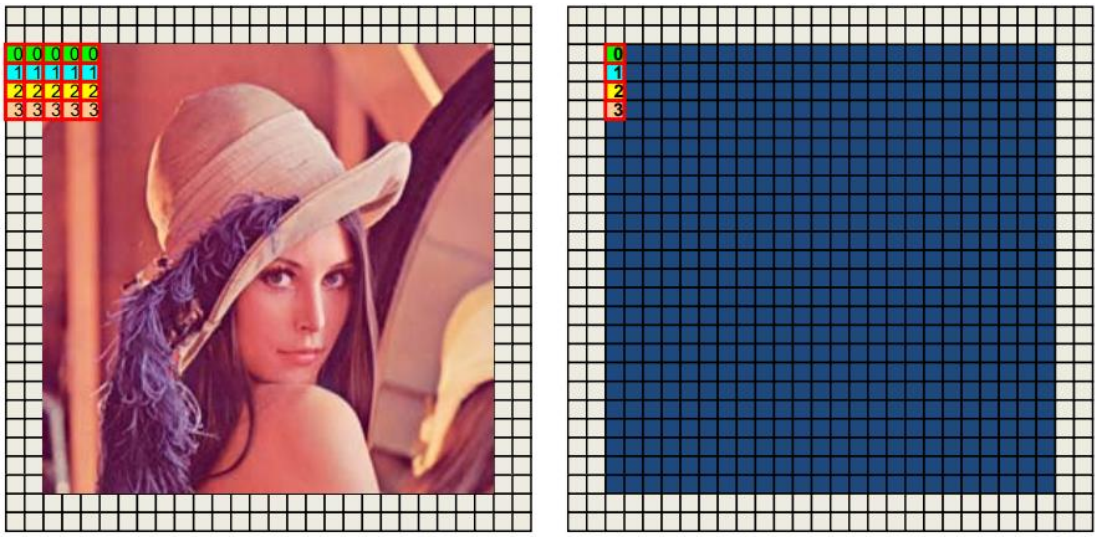


Рисунок 1 – Приклад розподілу даних вікном фільтру для потоків на одному блоці (вхідне зображення та результуюче у вигляді маски)

CUDA-пристрої обчислювальної здатності 2,0 або вище, можуть мати до 1024 потоків на блок, через деякі істотні вдосконалення апаратних засобів для CUDA.

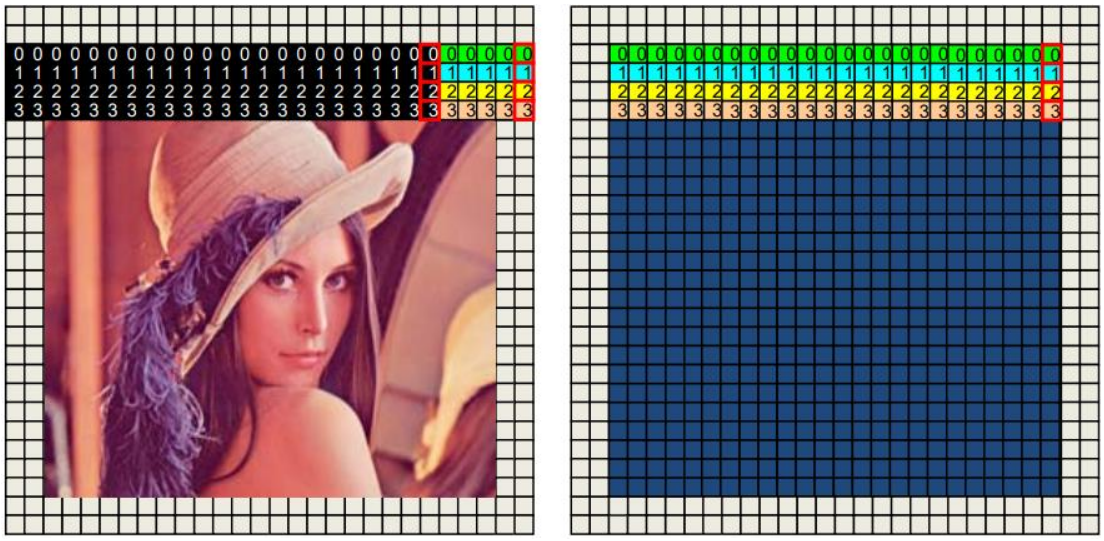


Рисунок 2 – Прохід вікном фільтру по ряду пікселів зображення

Для тесту було взято алгоритми розмиття та виявлення країв Гауса[5]. Час роботи ГП має значний приріст в порівнянні з процесорним

часом на всіх зображеннях, які були оброблені. Швидкість збільшення відсотка приросту швидкості зростає в міру збільшення розміру зображення. Це сходиться з більш раннім твердженням, що обробка CUDA є найбільш ефективною (рис. 3).

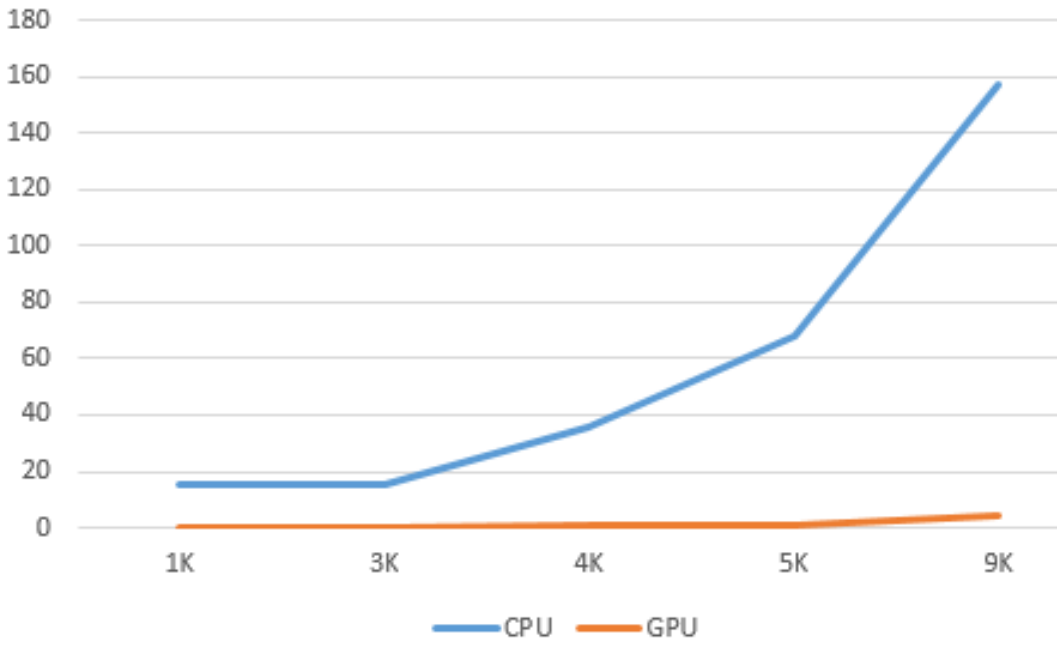


Рисунок 3 – Графік залежності часу виконання від розміру зображення

Графік підтверджує, що чим більші потужності потрібні для обчислення, тим більше CUDA буде корисна і тим більше ця технологія розкриє свій потенціал.

ВИСНОВКИ

У роботі надано короткий аналіз теоретичних основ білатеральної фільтрації та практичних основ застосування технології CUDA для цифрової обробки зображень. Приведені дані будуть застосовані для розробки прототипу паралельної реалізації алгоритму білатеральної фільтрації засобами мови Python. Приведений метод дозволить істотно підвищити швидкість виконання алгоритму на зображеннях великої роздільної здатності. Продукт, що розробляється, за істотного пришвидшення виконання, буде апробовано на системах реального часу за умови, якщо продукт не буде призводити до істотної втрати кадрів.

ЛІТЕРАТУРА

[1] **Tomasi C., Manduchi R.** Bilateral filtering for gray and color images // Proc. IEEE 6th Int. Conf. on Computer Vision - Bombay, India, Jan. 4-7, IEEE - 1998, - p.839 – 846.

[2] **Крапчатова Т. В., Филиппов М. В.** Анализ эффективности алгоритмов билатеральной фильтрации // Наука и образование - 77-30569/340957 - 2012.

[3] **Tae-Wuk Bae, Kyu-Ik Sohng** Small Target Detection Using Bilateral Filter Based on Edge Component // Infrared Milli Terahz Waves – 2010, Vol. 31 - p. 735–743.

[4] Fast Bilateral Filter – MIT [Електронний ресурс]. – Режим доступу: http://people.csail.mit.edu/sparis/bf/

[5] Image Processing with pyCUDA – Bones & Frames [Електронний ресурс]. – Режим доступу: http://www.marcogiordanotd.com/blog/python/image-processing-pycuda

[6] Intro to Image Processing with CUDA [Електронний ресурс]. – Режим доступу: http://supercomputingblog.com/cuda/intro-to-image-processing-with-cuda/

[7] Bilateral filtering definition – MIT [Електронний ресурс]. – Режим доступу:http://people.csail.mit.edu/sparis/bf_course/slides/03_definition_bf.pdf

[8] Fast Bilateral Filtering – Brown EDU [Електронний ресурс]. – Режим доступу: https://cs.brown.edu/courses/cs129/lectures/bf_course_Brown_Oct2012.pdf

[9] Fast Bilateral Filter – MIT [Електронний ресурс]. – Режим доступу: http://people.csail.mit.edu/sparis/bf/

Отримано 29.06.2017

УДК 004.8

РЕАЛІЗАЦІЯ ШТУЧНОГО ІНТЕЛЕКТУ У СТРАТЕГІЧНИХ ІГРАХ НА ПРИКЛАДІ ІГРОВОГО ДОДАТКУ «ШАХИ»

Глинська К.С.

Донецький національний технічний університет, Україна

Розглянута проблема реалізації штучного інтелекту в стратегічних іграх. Приведені алгоритми та евристики, що моделюють поведінку комп'ютерного шахового гравця. Протестовані й оцінені отримані результати, запропоновані подальші кроки по вдосконаленню ігрового штучного інтелекту.

Ключові слова: штучний інтелект, шахова програма, дерево рішень, альфа-бета відсікання, оцінка позиції.

1. ВСТУП

Одним з нових та широко застосовуваних напрямків науки є штучний інтелект. Майже кожного дня виникають нові розробки та технології, що функціонують на його основі та вирішують корисні для людини задачі. Штучний інтелект як наука включає багато відгалужень, серед яких виділяють моделювання міркувань людини, розпізнавання образів, інженерія знань та, зокрема, програмування інтелекту в комп'ютерних іграх.

Розробка ігрових додатків – це одна з поширених зараз галузей комп'ютерної індустрії, що активно розвивається, використовуючи для цього результати застосування все складніших технологій. Невід'ємною частиною багатьох сучасних ігрових додатків є реалізація штучного інтелекту. Під *ігровим штучним інтелектом* будемо розуміти сукупність програмних засобів, що використовуються в комп'ютерних іграх для моделювання розумної людської поведінки в діях персонажів, що управляються комп'ютером.

2. ПОСТАНОВКА ЗАДАЧІ

Багато параметрів сучасних ігрових додатків вже досягли вражаючих показників. Однак нові технології візуалізації й управління не компенсують один недолік сучасних комп'ютерних ігор – слабкий, далекий від людського способу сприйняття та поведінки штучний інтелект. В сучасних іграх штучний інтелект, як правило, поступається людині в ході прийняття рішень, поведінці персонажів в команді, оцінюванні подій, дії в нестандартних ситуаціях.

Створення штучного інтелекту – часто досить складні та нетривіальні задачі. Незважаючи на довгу історію розвитку цього напрямку досліджень, не існує єдиного алгоритму та погляду на програмування такого інтелекту у багатьох іграх, що пов'язано з існуванням різних підходів й комбінацій евристик, що обирає програміст на свій розсуд, та з неспроможністю сучасних комп'ютерів задовольнити всім вимогам розробника.

Дослідження та розробку штучного інтелекту зручно починати зі створення програмного модуля для гри в шахи, яку радянський математик О. С. Кронрод назвав «дрозофілою штучного інтелекту» [1]. Алгоритми та евристики, що використовують для імітації дій шахового гравця, є поширеними засобами реалізації штучного інтелекту в багатьох ігрових та інших додатках. Розуміння цих базових принципів програмування та поведінки комп'ютерного гравця дозволяє перейти до створення будь-якого штучного інтелекту, здатного аналізувати ситуацію, обирати стратегію поведінки залежно від її поточного стану та приймати найбільш сприятливе для нього рішення.

3. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

Більшість сучасних шахових програм можна віднести до однієї з трьох категорій:

- програми fast searchers;
- програми knowledge-based;
- гібриди між двома першими категоріями [2].

Ідея алгоритмів *fast searchers* полягає в тому, що, максимально спростивши функцію оцінки ходів, ретельно оптимізувавши всю програму в цілому, можна довести кількість позицій, що розглядаються програмою, до максимально високого числа. Алгоритм *knowledge-based*, навпаки, направлений на написання складної оціночної функції, у якій будуть враховуватись і взаємодія фігур, і безпека короля, і контроль важливих позицій на дошці.

Хоча більшість сучасних комп'ютерів мають досить високу продуктивність, покладатися лише на це було б нерозумно, враховуючи кількість розрахунків, які треба зробити навіть оптимізованій програмі. Математики вважають, що повне дерево перебору в шахах, хоча і є кінцевим, містить приблизно 10^{120} позицій [3]. Звичайно, формувати та аналізувати все дерево варіантів неможливо, але це і не потрібно. Шахіст рахує варіанти в середньому на 5-6 ходів, тому досить навчити програму будувати й оцінювати

ходи на обмежену глибину, знаходити цікаві й відкидати некорисні, відсікаючи зайві галузі дерева.

Дерево рішень – це спосіб представлення даних в ієрархічній, послідовній структурі, де кожному об'єкту відповідає єдиний вузол. В шахах кожному вузлу відповідає деякий хід гравця, рівні дерева – це послідовні ходи «білих» та «чорних», кожен шлях дерева, від вершини до листа, позначає деяку стратегію гри. Кожен лист дерева отримує оцінку – ступінь привабливості такого ходу для гравця. Рекурсивна обробка дерева рішень дозволяє обрати кращий поточний хід, проаналізувавши результати різних стратегій гри.

Алгоритм *мінімакс* базується на побудові дерева рішень та знаходженні вигіднішого ходу для комп'ютерного гравця, використовуючи отримані оцінки для комп'ютера та людини [4]. Спочатку, маючи множину припустимих на деякий етап гри ходів, формуються нові позиції фігур (перший рівень дерева). Після цього штучний інтелект аналізує можливі ходи суперника й «ходить» за нього (другий рівень дерева). Так продовжується до заданої глибини. Під час зворотної обробки дерева (вибору вузлів залежно від їх оцінки), алгоритм обирає шлях серед ходів з максимальними (для своїх ходів) та мінімальними (для ходів суперника) оцінками, припускаючи, що противник буде обирати розумні, вигідні для себе варіанти (тому гірші для комп'ютерного гравця). *NegaMax* – це аналог мінімаксу, але в даному алгоритмі оцінка позиції для однієї зі сторін дорівнює оцінці суперника з протилежним знаком (інвертована оцінка) [5].

Алгоритм *альфа-бета-відсікання* – це оптимізація алгоритму мінімакс, завдяки якій деякі гілки пошуку ігноруються. Основна ідея алгоритму полягає в наступному: якщо на один з наших ходів опонент має несприятливу для нас відповідь, то немає сенсу аналізувати інші його можливі відповіді на цей наш хід, тому що навіть якщо серед них і знайдуться більш сприятливі для нас, противник їх не вибере. Відсікання по альфі (ігнорування гіршого для нас ходу) також називають *Fail Low*, а відсікання по беті (ігнорування занадто гарної для нас стратегії, яку суперник не вибере) – *Fail High* [6].

Зрозуміло, що важним критерієм ефективності такого алгоритму є сортування ходів, тобто спочатку мають розглядались потенційно кращі з них. Відомим принципом сортування ходів в шахах є *MVV/LVA* (Most Valuable Victim – Least Valuable Aggressor, найбільш цінна жертва – найменш цінний нападаючий) [7]. Існують також додаткові евристики та прийоми, що дозволяють прискорити пошук кращого ходу за рахунок відсікання деяких гілок дерева рішень.

Найбільш ефективні з них були застосовані при реалізації шахового інтелекту.

Після генерації можливих ланцюжків ходів треба оцінити результат – поточну ситуацію на полі. Існує *матеріальна та позиційна оцінка*. Перша враховує лише кількість наявного матеріалу – фігур своїх та противника – з урахуванням цінностей кожного типу фігур. На позиційну оцінку впливає розташування фігури на полі відносно інших фігур та етапу гри [8].

4. РЕАЛІЗАЦІЯ ШАХОВОГО ІНТЕЛЕКТУ

В процесі розробки та тестування програмного додатку корегувалися алгоритми та їх параметри, оцінювався внесок окремих евристик у роботу штучного інтелекту. Наприклад, евристика Aspiration Search [9] поруч з іншими використаними евристиками не мала вагомого внеску, тому була опущена в кінцевому алгоритмі програми (див. рис.1).

Тестування вкладу різних евристик в продуктивність пошуку

	Null Move	NegaScout	PVS	Killer heuristic	History heuristic
Прискорення	56%	38%	74%	14%	17%

Результати гри розробленої програми з AsisChess

	Середній час ходу	Ходів у грі в середньому		Рівень ШІ
		За білих	За чорних	
myChessAI	3 сек.	63	32	Max (10)
AsisChess	30 сек.			Hard

Результати гри розробленої програми зі SparkChess

	Час ходу	Ходів у грі в середньому		Рівень ШІ
		За білих	За чорних	
myChessAI	3 сек.	22	51	Max
AsisChess	2-4 сек.			Max



Кінець гри за чорних проти AsisChess

Рисунок 1 – Результати тестування

В програмному засобі були реалізовані наступні евристики та алгоритми:

- алгоритм NegaMax та його оптимізація – альфа-бета відсікання для пошуку кращого ходу;
- алгоритм пошуку з нульовим вікном та NegaScout;
- евристика нульового ходу;
- PV-таблиці, Zobrist-ключи та хеш-таблиці для запам'ятовування кращих стратегій гри та оцінок позицій;
- алгоритм ітераційного занурення та алгоритм пошуку спокою при побудові дерева рішень;

- принцип MVV/LVA для сортування захоплень фігур;
- евристики вбивця та історії для сортування «тихих» ходів;
- алгоритми матеріальної та позиційної оцінки;
- подання дошки через бітборди, генератори ходів, атак, магічні бітборди [10].

В процесі та наприкінці розробки програми були проведені наступні види тестування:

- тестування працездатності програми в цілому, її відповідність поставленим вимогам до функціональності, швидкодії, інтерфейсу;
- тестування вкладу окремих евристик в роботу штучного інтелекту;
- тестування з різними параметрами (глибина дерева, обмеження часу пошуку, бонуси та штрафи за розташування фігур для отримання позиційної оцінки тощо) з метою пошуку оптимальних значень;
- тестування штучного інтелекту при грі з людиною та іншими програмами;
- тестування поведінки програми у заданих ситуаціях (класичні шахові задачі, тестові задачі на фігурну активність та ефект горизонту).

В результаті розробки було поставлено обмеження часу на хід від 1 до 3 секунд залежно від рівня складності. Максимальна глибина дерева також обмежується залежно від рівня – від 2 до 20 (10 рівнів складності). Як правило, загальний пошук альфа-бета не виходить за глибину 8 ходів при максимальної кількості фігур на полі, але «небезпечні» ходи (зокрема, взяття фігур) розглядаються (максимум – у 3 рази) глибше до деякої стабілізації ситуації.

Значно прискорила процес тестування база тестових наборів для тестування шахових движків, зараз відома як wacnew.epd, яку використовували та доповнювали ще з 70-х. Використана база тестів містила 300 позицій у форматі FEN та вказані кращі ходи для них. Розроблена програма успішно впоралася з тестами.

В цей же час треба враховувати, що програму ще можна вдосконалювати, підвищуючи рівень її штучного інтелекту до рівня кращих шахових програм. Зокрема, використання готових баз даних (бібліотек) дебютів та ендшпілей значно вплине на результат змагання з іншими програмами. Також можна не використовувати існуючі бази, а створити власні, змусивши движок навчатись на проведених іграх.

5. ВИСНОВКИ

В процесі виконання роботи були розглянуті основні поняття та принципи реалізації штучного інтелекту для гри в шахи, проаналізовані різні підходи до його розробки, запропонована

комбінація методів та евристик, що реалізує ефективну роботу штучного шахового інтелекту. На основі зібраного теоретичного матеріалу були спроектовані та розроблені програмні модулі, що реалізують штучний інтелект шахового гравця. Ігровий додаток відповідає сучасним вимогам та може бути використаний в реальних умовах. https://gitlab.cssum.net/tracker/tracker/tags/v0.6.3

У той же час було підтверджено складність навчання машини людському мисленню, неоднозначність інтерпретації різних ігрових ситуації професійним гравцем та штучним інтелектом, значна залежність останнього від потужностей техніки. Це змушує задуматися про пошук й використання додаткових, більш ефективних засобів реалізації штучного інтелекту у сучасних стратегічних іграх. Це можуть бути, наприклад, генетичні алгоритми, нейронні мережі, складніші функції оцінки та прийняття рішення, розпаралелювання процесів. Дослідження та впровадження цих прийомів у реалізацію штучного інтелекту стратегічної гри – наступний етап даної роботи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ:

[1] **Алимов А. А.** Искусственный интеллект в компьютерных играх. Многоуровневое планирование и реактивное поведение агентов / А. А. Алимов, О. А. Шабалина // *Известия Волгоградского государственного технического университета*. – 2011. – №10 (103). – С. 1–5.

[2] **Корнилов Е. Н.** Программирование шахмат и других логических игр / Е. Н. Корнилов. – СПб.: БХВ-Петербург, 2005. – 272 с.

[3] **Ботвинник М. М.** Алгоритм игры в шахматы / М. М. Ботвинник. – М.: Наука, 1968. – 94 с.

[4] **Ботвинник М. М.** Шах двадцатому веку / М. М. Ботвинник. – М.: Зебра Е, 2010. – 494 с.

[5] **Лысенко А. В.** Оценка позиции. Компьютерные шахматы / А. В. Лысенко, Е. Я. Гик. – М.: ФиС, 1990. – 176 с.

[6] **Джонс М. Т.** Программирование искусственного интеллекта в приложениях / М. Тим Джонс; Пер. с англ. – М.: ДМК Пресс, 2015. – 313 с.

[7] **Дамский Я. В.** Искусство шахмат. По законам красоты / Я. В. Дамский. – М.: Рипол Классик, 2005. – 152 с.

[8] **Волчок А. С.** Тактика и стратегия шахмат / А. С. Волчок. – К.: Здоровье, 1985. – 176 с.

[9] **Strejczek M.** Historical look on chess programming / M. Strejczek. – Telecommunication & Computer Science, 2004. – 142 с.

[10] **Levy D., Claude S.** Computer Chess Compendium / D. Levy, S. Claude. – Ishi Press, 2009. – 452 с.

Отримано 25.06.2017

УДК 004.5

РАЗРАБОТКА МОДУЛЯ СИМУЛЯЦИИ ИГРОВЫХ ПРИЛОЖЕНИЙ В СРЕДЕ UNITY НА БАЗЕ ТЕХНОЛОГИИ KINECT

А.С. Дзюбинский, О.А. Дмитриева
Донецкий национальный технический университет, Украина

Данная работа посвящена исследованию особенностей работы с технологией Kinect для отслеживания трехмерных движений человеческого тела и создания 3D-анимации в среде разработки игровых приложений Unity 3D. Выполнен обзор современных технологий захвата движений, освещены вопросы построения обучающей выборки для анализа и распознавания перемещений.

Ключевые слова: технология Microsoft Kinect, движение, Unity 3D, Sdk, Motion Capture

1. ВВЕДЕНИЕ

Microsoft Kinect – первый серьёзный шаг к созданию полноценной виртуальной реальности. В данной технологии реализуется интерактивное взаимодействие с пользователем путём оцифровывания движений его тела. На базе Kinect создаются приложения, позволяющие во время работы с ними проявить двигательную активность и полностью погрузиться в увлекательный процесс управления приложением или игрой. Программ, адаптированных под Ms Kinect, на данный момент существует не так много, и далеко не всем разработчикам удалось в полной мере освоить умение создавать приложения под эту сложную систему дополненной реальности [1].

Kinect – это набор аппаратных и программных возможностей для отслеживания движений в 3D пространствею. Представляет собой небольшую коробку с двумя камерами и инфракрасным излучателем:

- IR Emitter – Инфракрасный излучатель. Его назначение испускать инфракрасные лучи, которые, отражаясь от предметов, попадают назад в сенсор, где их принимает:
- IR Depth Sensor – Инфракрасный приёмник. Собирает отраженные лучи, преобразуя их в расстояние от сенсора до

объекта(ов). И таким образом строится матрица расстояний — целый кадр. Максимальное разрешение 640x480 (30 fps).

- Color Sensor – Цветная камера. Захват видео с максимальным разрешением 1280x960 (12fps). Угол обзора камеры: 43° по вертикали и 57° по горизонтали (угол обзора дальномера идентичный). Можно выбрать формат картинки: RGB или YUV.
- Microphone Array – набор микрофонов. 4 встроенных микрофона позволяют определить местоположение источника звука и направление звуковых волн. Встроенный обработчик звукового сигнала включает подавление эха и уменьшение шума.
- Tilt Motor – Коррекция наклона. Позволяет программно настраивать наклон сенсора в диапазоне ±27° по вертикали [2,3].

Контроллер Microsoft Kinect позволяет управлять игрой с помощью тела, сенсор отслеживает каждое движение. При этом задействованы все части тела: руки, ноги, колени, талия, бедра. Сенсор Kinect улавливает движения при перемещении влево или вправо, при прыжках или приседаниях. Компания Microsoft предоставляет инструментарий для работы с Kinect – Kinect for Windows SDK, рассчитанный для всех кто заинтересован в изучении возможностей NUI (Natural User Interface) с помощью Kinect и связанных технологий [4].

2. ПОСТАНОВКА ЗАДАЧИ

Процесс разработки игр с помощью технологии распознавания движений не приобрел большой популярности у разработчиков игр, к тому же многие среды разработки начали поддерживать эту технологию только спустя годы, что привело к падению интереса к Kinect. В связи с растущей популярностью AR, VR, которые позволяют на данном этапе технологий отслеживать движения в пространстве только головы, технология Kinect может помочь в отслеживании остальных частей тела, что в совокупности даст полное погружение в виртуальную реальность. Особенно востребованы спортивные и танцевальные игры, комплексы тренировок, приложения для йоги и фитнеса, обучающие симуляторы.

3. ЦЕЛЬ РАЗРАБОТКИ И ЗАДАЧИ

Целью данной работы является создание игрового приложения с использованием технологии Kinect Motion Capture в среде разработки игр Unity 3d для возможного дальнейшего его использования как

основы для любой Kinect игры, что существенно упростит и ускорит процесс разработки подобных игр. Поставленные задачи:

- провести обзор существующих технологий захвата движений, выявить их особенности и сравнить с технологией Microsoft Kinect.
- провести анализ обучающей дискретной выборки для анализа и распознавания Motion Capture движений. Обученная модель должна определять сам факт наличия жеста (возвращая степень уверенности — был ли в данный момент времени жест).
- изучить особенности программирования на базе устройства Kinect в частности ознакомиться с комплектом средств разработки Kinect for Windows SDK, Unity SDK и интегрированной средой разработки игр Unity 3D;
- выполнить проектирование, реализацию, отладку и тестирования простого программного продукта с использованием технологии Kinect в среде разработки Unity для возможности его дальнейшего использования как базового шаблона при создании любых игр на данной платформе;
- ознакомиться с другими возможностями Kinect и сторонних SDK – распознавание речи с помощью Skeletal Tracking (трекинг человеческой фигуры), Face Tracking (трекинг лица), Speech Recognition (распознавание речи) [5].

ЛІТЕРАТУРА

[1] **Jarrett Webb, James Ashley**, Beginning Kinect Programming with the Microsoft Kinect SDK (Expert's Voice in Microsoft), 1st ed. Edition, 2012, pp. 324.

[2] **Clemente Giorio, Massimo Fascinari**, Kinect in Motion – Audio and Visual Tracking Example, April 2013, 430p

[3] Unity 5 and kinect 2 integration [Электронный ресурс]. – Режим доступа: http://www.imaginativeuniversal.com/blog/2015/03/27/ unity-5-and-kinect-2-integration/

[4] Пользовательские жесты, Kinect + Unity [Электронный ресурс]. – Режим доступа: https://habrahabr.ru/company/ singularis/blog/276455/

[5] Microsoft Kinect - Microsoft SDK [Электронный ресурс]. – Режим доступа: http://wiki.etc.cmu.edu/unity3d/index.php/ Microsoft_Kinect_-_Microsoft_SDK

Отримано 27.06.2017

УДК 004.94

БАЗОВІ ПІДХОДИ ДО РЕАЛІЗАЦІЇ НАВІГАЦІЇ У ТРЬОХВИМІРНИХ ВІДЕОІГРАХ

О.М. Кривенко, С.А. Зори
Донецький національний технічний університет,
м. Покровськ, Україна

В роботі надано короткий аналіз існуючих підходів до засобів та приведено приклади їх реалізації.

Ключові слова: навігація, відеогра, векторна навігація, NavMesh, Waypoints.

1. ВСТУП. АКТУАЛЬНІСТЬ

Сучасна комп'ютерна індустрія має достатньо велику кількість засобів для реалізації задач, котрі постають при розробці відеоігор. Однією з таких задача створення штучного інтелекту. Це дуже складна задача, оскільки розробник повинен надати інтелекту здатність орієнтуватися в ситуації та приймати дії, що відповідають до певній ігровій ситуації. Наприклад, патрулювання місцевості, переслідування гравця, або інше. У багатьох відеоіграх невідмінними від штучного інтелекту є алгоритми відповідні за пересування гравців. Ці алгоритми можуть бути представлені, як маленьким, так і величезним алгоритмом. Але всі алгоритми базуються на одних базових підходах. Деякі з таких алгоритмів входять до комплекту постачання ігрових рушіїв.

2. ВИЗНАЧЕННЯ МАРШРУТІВ ЗА СКЛАДНІСТЮ

За складністю маршрути переміщення можна віднести до простих та складних. Прості маршрути мають одну фінальну точку і переміщення здійснюється від початкової точки розміщення до фінальної. У складних маршрутах окрім фінальної точки є декілька точок, які послідовно проходяться до досягнення фінальної точки. Складний підхід до навігації, ще називають WayPoints.

3. БАЗОВІ ПІДХОДИ ДО ПОБУДОВИ МАРШРУТІВ

3.1 Алгоритм векторного руху

Це найпростіший алгоритм руху. Він реалізується за допомогою по шаговій зміні координат до координати фінальної точки маршруту. Може бути використаний при переміщенні, якщо на путі не

зустрічається перешкоди, які повинні бути враховані. Є найбільш швидкодіючим з алгоритмів та не вимагає додаткової пам'яті та додаткових обчислень для здійснення руху.

3.2 Алгоритм руху за ломаною або дугою

Це модифікація векторного алгоритму, що призвана викоренити недолік векторного алгоритму руху в вигляді не врахування перешкод. Тобто в результатом роботи даного алгоритму є маршрут який враховує всі перешкоди.

Алгоритм пересування за ламаною заснований на пошуку шляху за допомогою визначення крайніх точок перешкоди, та додання до маршруту проміжної точки або декількох точок руху, щоб було реально обійти маршрут. Маршрут за ламаною має недолік в тому, що інколи в крайніх точках маршруту пересування буде виглядати не природньо. Такої особливості позбавлений алгоритм руху за дугою, де пересування здійснюється не по прямій лінії, а за дугою, що дозволяє зробити пересування персонажа більш природнім.

Дані алгоритми мають загальний недолік в тому, що у випадку великої кількості параметрів для прорахунку вони мають суттєву складність.

3.3 Рух за навігаційною сіткою

Навігаційна сітка може використовуватися, як для зменшення складність алгоритмів руху за ломаною та дугою, так і для алгоритму руху розробленої саме для неї.

В випадку з алгоритмами пересування за ломаною та дугою вона дозволяє вираховувати межі перепон кожен раз, що суттєво зменшує навантаження.

Навігаційна сітка або як її ще називають NavMesh представляє собою спосіб представлення даних про шляхи на ігровій карті. Вона з себе представляє полігональною поверхнею, яка покриває ті частини ігрового ландшафту, де можуть пройти наші персонажі.

Полігони «NavMesh» будуються таким чином, щоб вийшли опуклі примітиви. Опуклість примітиву дає нам гарантію того, що всі об'єкти, що знаходяться всередині цього багатокутника доступні нам без жодних перешкод. Це одне з основних властивостей «NavMesh», і всі спеціалізовані алгоритми пошуку шляху через «NavMesh» активно використовують цей постулат.

Завдяки своїй природі, алгоритми, побудовані на базі «NavMesh», є дуже швидкими. Це обумовлено відсутністю необхідності застосовувати трасування променю (англ. RayCast) для пошуку перешкод. До переваг даного алгоритму відноситься зниження витрати пам'яті на зберігання графа шляхів, більш природне переміщення по карті, на відміну від незграбних переходів інших алгоритмів.

ВИСНОВКИ

Сучасна ігрова індустрія надає багато засобів для реалізації ігрової навігації. Всі методи мають свої недоліки та свої переваги відповідно до своєї області використання. Але серед методів можна виділити навігацію за навігаційної сіткою. Зараз цей метод де-факто став базовим для більшості ігрових рушіїв. Оскільки він є найбільш оптимальним рішення для вирішення більшості задач.

ЛІТЕРАТУРА

1. **Brian Moakley**. Unity Games by Tutorials: Make 4 Complete Unity Games from Scratch Using C# / Brian Moakley, Mike Berg and other. – Apress, 2016. – 352 Pages.
2. **Jere Miles**. Unity 3D and Playmaker Essentials: Game Development from Concept to Publishing / – CRC Press, 2016. – 482 Pages.
3. **Joe Hocking**. Unity in Action: Multiplatform Game Development in C# with Unity 5 / Joe Hocking. – Manning, 2015. – 327 Pages.
4. **Alan Thorn**. Mastering Unity Scripting / Alan Thorn. – CRC Press, 2015. – 482 Pages.

Отримано 29.06.2017

УДК 004.9

МЕТОДИ ПОБУДОВИ ВІРТУАЛЬНОГО ПРЕДМЕТНО-ПРОСТОРОВОГО РОЗВИВАЮЧОГО СЕРЕДОВИЩА

Ю. А. Бишенко, Н. С. Костюкова, Г. В. Тищенко
Державний вищий навчальний заклад «Донецький національний технічний університет»

Були надані визначення ключовим поняттям: «розвиваючі ігри», «ігрове навчання», «навчаючі відеоігри». Розглядається класифікація існуючих розвиваючих відеоігор та їх застосування. Було проаналізовано ігри жанру «Brain fitness» та виявлено їх основні функції.

Ключові слова: ігрове навчання, навчальні відеоігри, освітня гра розвиваюча гра, brain fitness video game.

1. ВСТУП

Ігрове навчання (Educational entertainment) - це форма навчального процесу в умовних ситуаціях, спрямована на відтворення і засвоєння суспільного досвіду у всіх його проявах: знаннях, навичках, уміннях, емоційно-оціночної діяльності. В даний час часто називається едьютейнментом (англ. Edutainment), яке означає навчальну розвагу або навчання за допомогою розваги. [1-3]

2. ЗАГАЛЬНА ХАРАКТЕРИСТИКА НАВЧАЛЬНИХ ВІДЕОІГОР

Навчальні відеоігри - відеоігри, розроблювані для навчання людей будь-якого віку наданням інформації стосовно якогось об'єкту чи об'єктів, зміни уявлень про ці об'єкти, популяризації ідей та концептів, навчання або допомоги у вивченні визначених навичок. Навчальні відеоігри у процесі гри можуть познайомити дитину з тваринним світом або навчити рахувати, а дорослому допомогти вивчити іноземну мову чи розказати про якісь політичні події, наприклад вибори президента.

Важливою частиною навчальної гри є зручний та зрозумілий інтерфейс, легкість інсталяції або можливість грати без неї. Якщо гра призначена для браузерів та гри по інтернету, важливою є сумісність з популярними браузерами та підтримка повільних з'єднань.

Так само як і в навчальній літературі та відео, у відеоіграх часто використовуються персонажі-викладачі, помічники та аватари.

Наприклад, гравець може взяти на себе роль казкового персонажа, або таким персонажем буде помічник. Персонажами також можуть виступати відомі історичні особи, тварини, герої мультфільмів та художніх творів. [4-6]

Освітня гра (Розвиваюча гра, Educational game) - це вид гри, розроблений з метою кращого засвоєння навчального матеріалу за допомогою ігрових елементів. Освітні ігри покликані допомогти людині у вивченні конкретних об'єктів, розробці концепцій, осмисленні історичних подій, культурного розвитку, а також освоєнні нових вмінь в ході ігрового процесу.

Класифікація розвиваючих відео ігор представлена на рисунку 1. [7]

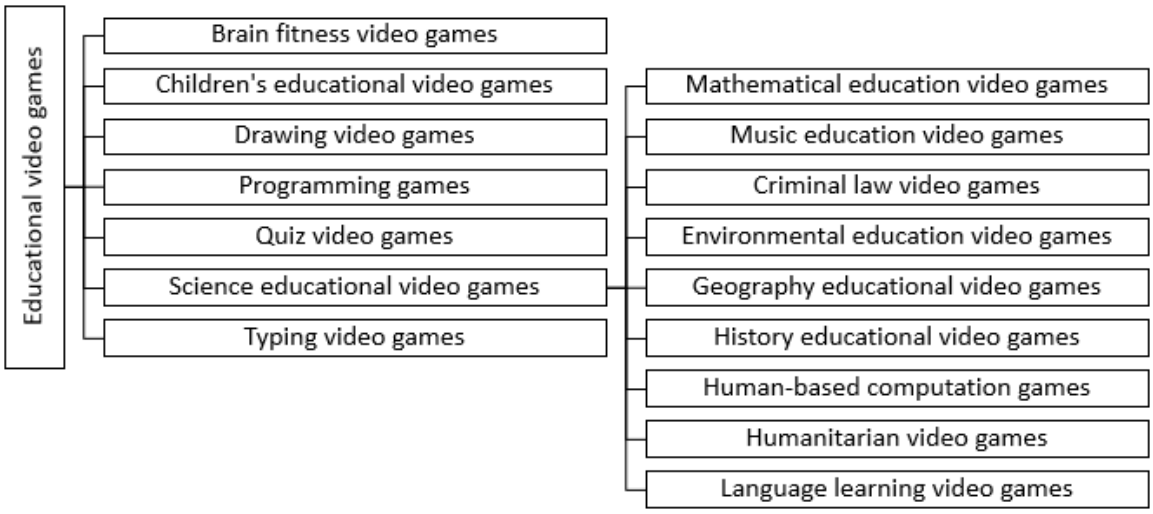


Рисунок 1 – Класифікація розвиваючих ігор

Brain fitness video games (cognitive training або neurobics) – ігри цієї групи побудовані на гіпотезі, що когнітивні здатності можливо підтримувати або поліпшити за допомогою «вправ для мозку», аналогічно до фізичних вправ, які поліпшують стан тіла.

Хоча є переконливі докази того, що деякі аспекти структури мозку залишаються пластичними на протязі всього життя, і що високі рівні розумової діяльності пов'язані зі зменшенням ризику вікової деменції, наукова підтримка поняття «фітнес мозку» обмежена. Термін практично ніколи не використовується в науковій літературі, але зазвичай використовується в контексті самодопомоги книг і комерційних продуктів. [8]

Для порівняння було обрано наступні ігри у жанрі Brain fitness video games: Big Brain Academy, Brain Age, Brain Challenge, Mind Quiz. В таблиці 1 наведені результати порівняння ігор та на рисунку 2 наведена схема переходів між екранами в проаналізованих іграх.

Таблиця 1 – Типові функції в Brain fitness іграх

Критерії	Big Brain Academy	Brain Age	Brain Challenge	Mind Quiz
Градація навичок (скільки/котрі)	Think (мислення), Analyze (судження), Compute (математичні), Identify (візуальні), Memorize (пам'ять)	-	Logic (логіка), Math (математика), Visual (візуальні), Focus (уважність)	Calculation (математичні), Reflex (реакція), Judgment (судження), Memory (пам'ять)
Градація рівнів важкості головоломок	+ (3)	-	+ (3)	-
Режими гри (кількість)	3	3	2	7
Вхідне тестування навичок	-	+	-	+
Бистра гра	+	+	+	+
Щоденне тренування	+	+	+	+
Онлайн змагання	+	-	-	+
Збір Статистики	+	+	+	+

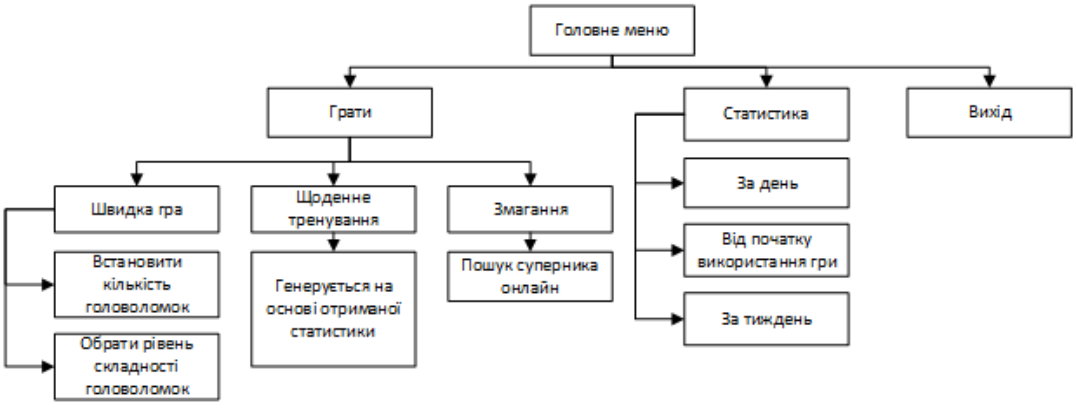


Рисунок 2 – Схема переходів

3. ВИСНОВКИ

В ході аналізу були виявлені типові риси, притаманні більшості brain fitness відеоігор, а саме:

- градація головоломок за навичками, котрі ті поліпшують;
- наявність вибору рівня складності для головоломок.
- наявність декількох режимів гри – одиночна гра, мережеве змагання, та бистре тренування.
- можливість відстежувати статистику прогресу гравця.

ЛІТЕТАТУРА:

1. Игровое обучение [Електронний ресурс]: «Википедия». – Режим доступу: https://ru.wikipedia.org/wiki/Игровое_обучение – Дата доступу: квітень 2017. – Загол. з екрану.
2. Educational entertainment [Електронний ресурс]: «Wikipedia». – Режим доступу: https://en.wikipedia.org/wiki/Educational_entertainment – Дата доступу: квітень 2017. – Загол. з екрану.
3. What is Edutainment? [Електронний ресурс]: «WhatIs». – Режим доступу: http://whatis.techtarget.com/definition/edutainment – Дата доступу: квітень 2017. – Загол. з екрану.
4. Educational video game [Електронний ресурс]: «Wikipedia». – Режим доступу: https://en.wikipedia.org/wiki/Educational_video_game – Дата доступу: квітень 2017. – Загол. з екрану.
5. Обучающая игра [Електронний ресурс]: «Википедия». – Режим доступу: https://ru.wikipedia.org/wiki/Обучающая_игра – Дата доступу: квітень 2017. – Загол. з екрану.
6. Навчальна відеогра [Електронний ресурс]: «Вікіпедія». – Режим доступу: https://uk.wikipedia.org/wiki/Навчальна_гра – Дата доступу: квітень 2017. – Загол. з екрану.
7. Category:Educational video games [Електронний ресурс]: «Wikipedia». – Режим доступу: https://en.wikipedia.org/wiki/Category:Educational_video_games – Дата доступу: квітень 2017. – Загол. з екрану.
8. Cognitive_training [Електронний ресурс]: «Wikipedia». – Режим доступу: https://en.wikipedia.org/wiki/Cognitive_training – Дата доступу: квітень 2017. – Загол. з екрану.
9. J. P. Gee. What Video Games Have to Teach Us About Learning and Literacy – USA: Palgrave Macmillan, 256 c. – 2007
10. J. Schell. The art of game of design. – USA: Morgan Kaufmann Publishers, 518 c. - 2008

Отримано 26.06.2017

УДК 004.4

ЛОГІЧНА ГРА- ГОЛОВОЛОМКА З ВИКОРИСТАННЯМ ДЕДУКТИВНОГО МЕТОДУ ПОШУКУ У БАЗІ ДАНИХ

О. Ю. Лукін, Н. С. Костюкова
Донецький національний технічний університет, м. Покровськ, Україна

Представлено та розглянуто опис гри-головоломки «Шерлок Холмс», в основі якої лежить розв'язання загадки Ейнштейна. Було розроблено програмний продукт - логічну гру, проаналізовано вимоги до неї та протестовано реалізовану програму.

Ключові слова: гра-головоломка, загадка Ейнштейна, метод пошуку.

1. ВСТУП

Дедуктивний метод пошуку в базі даних полягає в тому, що відомі співвідношення записуються до таблиці, при цьому неможливі варіанти послідовно вилучаються. Даний метод лежить в основи кількох логічних ігор- головоломок. Однією з таких ігор є "Sherlock the logic game". Вона вчить аналізувати інформацію та логічно міркувати. Традиційно логічні ігри на базі загадки Ейнштейна полягають у заповненні таблиці з певною інформацією, розмірність якої залежить від версії гри і складності, використовуючи випадково сформовані підказки. Після вірного заповнення гра закінчується. Нарахування балів в різних версіях ігор досить різноманітне і ґрунтується на характеристиках часу, кількості зроблених помилок та використанні додаткових підказок.

2. ЗАГАДКА ЕЙНШТЕЙНА

Загадка Ейнштейна – відома логічна задача, за легендою створена Альбертом Ейнштейном в роки його дитинства. Також існує думка, що вона використовувалася Ейнштейном для перевірки кандидатів в асистенти на здатність до логічного мислення. Іноді автором головоломки називають Льюїса Керролла [2].

У своєму найскладнішому варіанті завдання передбачає вирішення в розумі, без використання будь-яких записів або засобів збереження інформації. Без цих обмежень головоломка помітно втрачає в складності, оскільки може бути вирішена простим складанням таблиці з вилученням явно суперечливих варіантів, і, отже, мало що говорить про здібності випробуваного.

На рисунку 1 можна побачити текст загадки.

Рішення цієї загадки полягає у тому, щоб вписати відомі співвідношення в таблицю, послідовно виключаючи неможливі варіанти [4].

Оригінальний текст загадки Ейнштейна (переклад з англ.)

На вулиці розташовані п'ять будинків.
Англієць проживає в будинку червоного кольору.
Іспанець тримає собаку.
У будинку зеленого кольору п'ють каву.
Українець любить чай.
Будинок зеленого кольору розташований відразу праворуч від будинку білого кольору.
Людина, що палить «Old Gold» розводить равликів.
У будинку жовтого кольору живе палить «Kool».
У будинку по центру п'ють молоко.
Норвежець живе в будинку №1.
Сусід кращого «Chesterfield» тримає в будинку лисицю.
У будинку, сусідньому з тим, де тримають коня, живе палить «Kool».
Людина, що палить «Lucky Strike» любить апельсиновий сік.
Японець курить «Parliament».
Норвежець живе в будинку, розташованому поруч з будинком синього кольору.

Питання: Хто п'є воду, а хто тримає зебру?

Рисунок 1 – Текст загадки Ейнштейна

"Sherlock the logic game" – головоломка американського інді–розробника Еверетта Кейзера, яка була створена у 1991 році. Ідея гри була взята з класичної загадки Ейнштейна. Гравець виступає в ролі Шерлока Холмса, його мета - скласти вірну таблицю фактів, спираючись на ті підказки, які йому дано.

3. ОПИС ПРОГРАМНОГО ДОДАТКУ SHERLOCK.

Структура ігрового додатку складається з наступних модулів: модуль меню та модуль для роботи гри. На рисунку 2 можна побачити структуру ігрового додатку.

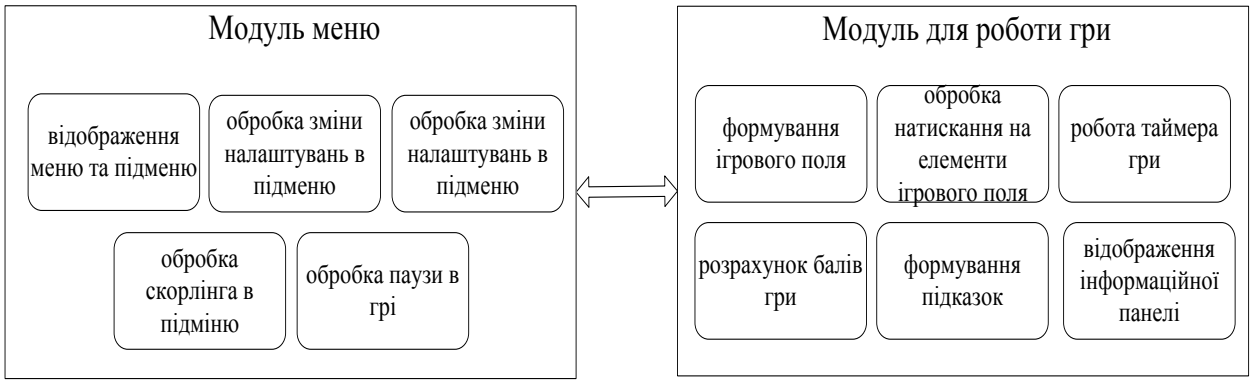


Рисунок 2 – Структура ігрового додатку

Модуль меню виконує всю роботу по функціонуванню меню та підменю і включає:

– відображення меню та підменю;
– обробку натискань елементів меню;
– обробку зміни налаштувань в підменю;
– обробку скролінга в підміню;
– обробку паузи в грі.

На рисунку 3 показана сцена головного меню і ієрархія об'єктів, розташованих на цій сцені.



Рисунок 3 – Сцена головного меню і ієрархія об'єктів

На сцені головного меню присутні неактивні елементи – елементи підменю, вони стають активними, коли користувач заходить у це підменю.

При створенні цієї сцени були використані майже усі елементи інтерфейсу користувача Unity: Canvas, Panel, Image, Scrollbar, Slider, Text, Toggle, Button.

Для роботи головного меню було створено два скрипти – це MenuScript та OptionsScript. MenuScript відповідає за роботу головного меню, а OptionsScript – за роботу налаштувань. Ці скрипти містять у собі класи з їх назвою. Так, у MenuScript міститься клас MenuScript, а у OptionsScript – клас OptionsScript.

Модуль для роботи гри виконує головну функцію додатку, він забезпечує роботу гри.

Модуль для роботи гри включає:
– відображення ігрового поля;
– відображення інформаційної панелі.
– обробку натискання на елементи ігрового поля;
– роботу таймера гри;
– розрахунок балів гри;
– формування підказок.

На рисунку 4 показана сцена гри зі складністю 4x4 і ієрархія об'єктів, розташованих на цій сцені.

Для реалізації модулів використано механізм сцен Unity. В розробленому додатку в Unity було створено 4 сцени:

– «Menu» – являє собою головне меню гри;

– «Game4» – являє собою ігровий простір, на якому безпосередньо відбувається весь ігровий процес, зі складністю гри 4x4;

– «Game5» – являє собою ігровий простір, на якому безпосередньо відбувається весь ігровий процес, зі складністю гри 5x5;

– «Game6» – являє собою ігровий простір, на якому безпосередньо відбувається весь ігровий процес, зі складністю гри 6x6.



Рисунок 4 – Сцена гри і ієрархія об'єктів

Для роботи гри було створено п'ять скриптів: PauseScript, Puzzle, Clues, SetupPuzzle та TimerScript. PauseScript відповідає за роботу паузи. SetupPuzzle відповідає за роботу гри. Puzzle та Clues – це допоміжні скрипти SetupPuzzle. TimerScript відповідає за роботу таймера гри.

Загалом програма складається з семи класів.

На рисунку 5 зображена діаграма класів, на якій показано загальну структуру програми.

MonoBehaviour – це базовий клас для роботи з Unity, від якого успадковуються скрипти.

Клас MenuScript відповідає за роботу головного меню, клас PauseScript відповідає за роботу паузи у грі, клас TimerScript відповідає за роботу таймера, клас OptionScript відповідає за роботу меню опцій, клас SetupPuzzle відповідає за роботу гри. Клас Puzzle та Clues є допоміжними класами для класу SetupPuzzle.

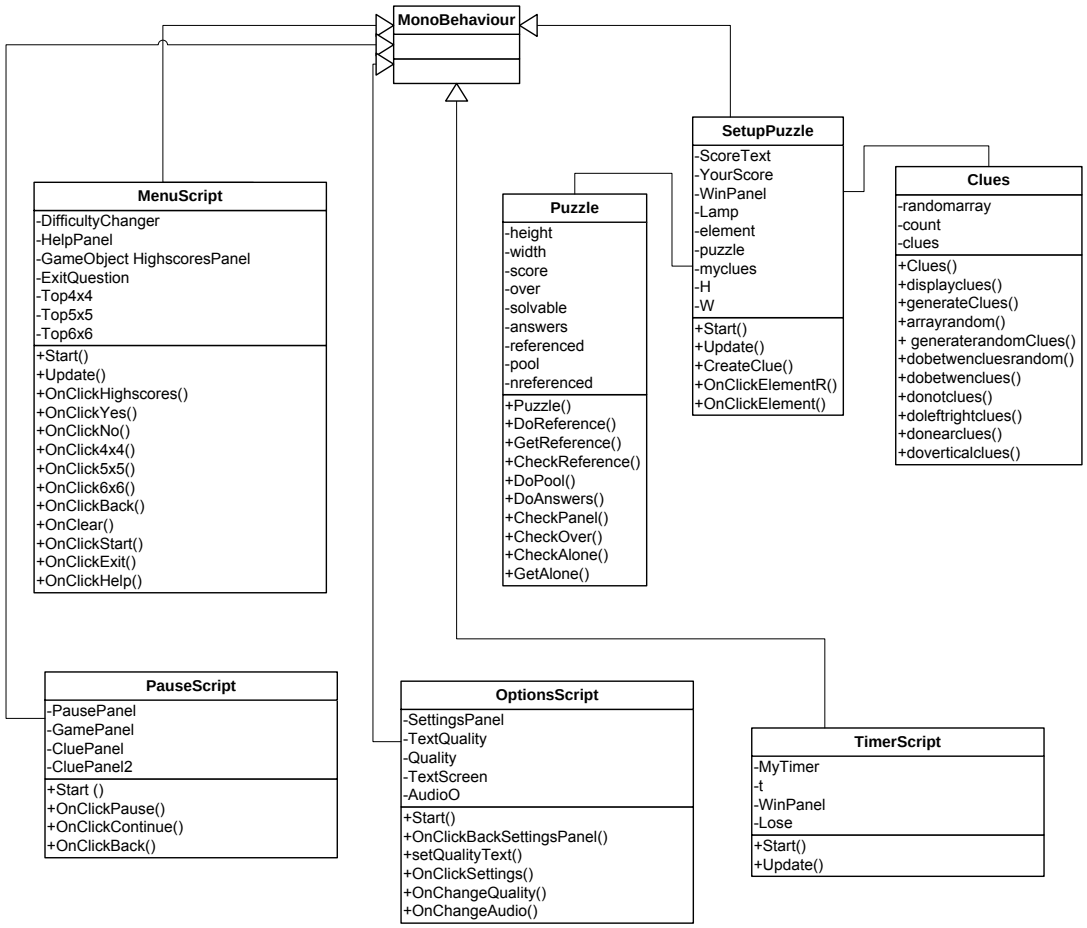


Рисунок 5 – Діаграма класів

ВИСНОВКИ

Описаний програмний додаток є працездатною логічною комп'ютерною грою, створену засобами мови програмування C # та ігрового движка Unity, яка має зручний і інтуїтивно зрозумілий інтерфейс, дозволяє використовувати різну складність вхідної таблиці, має таблицю рекордів.

ЛІТЕРАТУРА**:**

1. Паркин С. Самые знаменитые компьютерные игры / С. Паркин; [пер. з англ. М. А. Райтмана]. - Москва : Эксмо, 2015. - 255 с.
2. Sherlock the logic game [Електронний ресурс]. – Режим доступу: http://www.kaser.com
3. Загадка Эйнштейна [Електронний ресурс]. – Режим доступу: https://ru.wikipedia.org/wiki/Загадка_Эйнштейна
4. Stangroom J. Einstein's Riddle: Riddles, Paradoxes, and Conundrums to Stretch Your Mind / J. Stangroom - USA: Bloomsbury, 2009.-144 c.

Отримано 29.06.17

УДК 004.92

ІНТЕРАКТИВНА ВІЗУАЛІЗАЦІЯ ЗВ'ЯЗКІВ У РЕЛЯЦІЙНІЙ БАЗІ ДАНИХ

Н.С.Костюкова, Є.О.Башков, А.А.Матрьонін

Донецький національний технічний університет, м. Покровськ

В роботі було виконано порівняння запропонованого авторами модифікованого алгоритму візуалізації графів з існуючими рішеннями, які використовуються на реальних проектах. Порівняння виконувалось на основі естетичних критеріїв мінімізації: загального числа перетинів ребер, розмірів областей, загальної кількості ребер та загальної довжини ребер.

Ключові слова: граф, візуалізація, естетичні критерії мінімізації.

1. ВСТУП

Задача візуалізації схеми реляційної бази даних є однією з головних потреб розробників баз даних, яка виникає в процесі проектування програмного продукту. Схема бази даних являє собою граф, який відображає всі сутності і зв'язки між цими сутностями. Візуалізація графа може бути більш або менш вдалою в залежності від алгоритму, за яким виконується. Візуалізація схеми реляційної бази даних ускладнюється тим, що така схема являє собою граф з вершинами довільної форми, які мають підписи. Дана робота описує модіфікацію алгоритму інтерактивної візуалізації бази даних з дотриманням загальноприйнятих естетичних критеріїв. Описаний в роботі алгоритм є розробкою авторів і задовольняє трьом з чотирьох естетичним критеріїв.

Результати можуть бути використані на будь-яких реляційних базах даних. Новизна даної роботи полягає в посиленні критерію мінімізації перетинів між зв'язками таблиць.

2. ПОСТАНОВКА ЗАДАЧІ ВІЗУАЛІЗАЦІЇ СХЕМИ БАЗИ ДАНИХ

Схема бази даних має вигляд графа, кожна вершина якого описує таблицю цієї бази даних, а ребро – зв'язок між двома таблицями. У загальному значенні граф представляється як набір вершин (вузлів), з'єднаних ребрами. У строгому визначенні графом

називається така пара множин G = (V, E), де V є підмножина будь-якої зліченної множини, а E - підмножина зв'язків V з V [1].

Нехай задана деяка модель бази даних, потрібно розробити алгоритм, який на основі сукупності абстрактних об'єктів моделі (таблиць) будує двовимірне зображення графа у вигляді їх геометричних образів, представляючи наявні в ній таблиці, їх стовпці, ключі та зв'язки.

Проведемо формалізацію описаної задачі, для чого введемо необхідні параметри відображення, які повинні бути використані для порівняння існуючих рішень з розробленою модифікацією алгоритму:

– загальна кількість перетинів ребер (цей параметр має бути мінімізований; в ідеалі, якщо це можливо, повинно бути отримано планарне відображення);

– розмір областей - кількість даних, які ми можемо вмістити на одиницю площі (цей параметр має бути мінімізований;);

– загальна довжина ребер (необхідна мінімізація сумарної довжини всіх ребер, це впливає на цілісність зв'язків);

– універсальна довжина ребер (необхідна мінімізація відмінностей у довжинах ребер).

Слід відзначити, що критерій мінімізації загальної довжини ребер дуже сильно конфліктує з критерієм мінімізації розмірів області.

3. ОГЛЯД ПІДХОДІВ ДО ВІЗУАЛІЗАЦІЇ В ІСНУЮЧИХ КОНСТРУКТОРАХ БАЗ ДАНИХ

Конструктор баз даних - це програмний засіб, що дозволяє конструювати і візуалізувати базу даних, з якою встановлено з'єднання. Конструктор баз даних використовується для створення, редагування і видалення таблиць, атрибутів (стовпців), ключів, індексів, зв'язків і обмежень. Для візуалізації бази даних можна створити одну або декілька діаграм [2]. Будь- яка сучасна система управління базами даних (СУБД) має у своєму складі конструктор баз даних.

Для будь-якої бази даних можна створити будь-яку необхідну кількість діаграм; кожна з таблиць бази даних може використовуватися в будь-якій кількості діаграм. Таким чином, для візуалізації різних частин бази даних або для акцентування різних аспектів її конструювання можна створювати різні діаграми. Наприклад, можна створити велику діаграму, в якій будуть відображатися всі таблиці і стовпці, а також меншу діаграму, в якій

будуть відображатися всі таблиці, але не буде стовпців. Приклад конструктора бази даних наведений на рисунку 1.16.

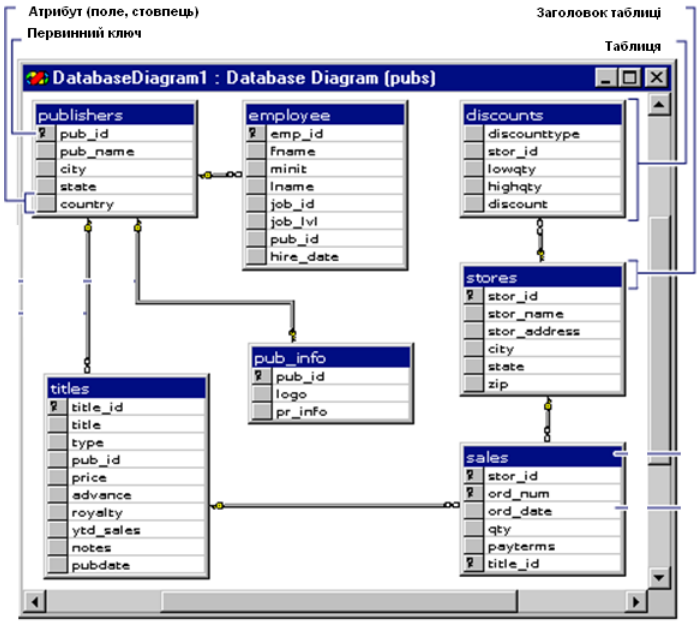


Рисунок 1 - Приклад візуалізованої схеми бази даних

Кожна створена діаграма зберігається у відповідній базі даних. Усередині діаграми бази даних у кожного зв'язку є три окремих елементи: кінцеві точки, стиль лінії і пов'язані таблиці.

Кінцеві точки лінії показують вид зв'язку: «один до одного» або «один до багатьох». Якщо у зв'язку по одному ключу на кожній кінцевій точці, то це зв'язок «один до одного».

Стиль лінії – це різновид лінії (суцільна, пунктирна тощо), за допомогою якого показується, чи перевіряє СУБД цілісність посилань при додаванні нових даних в таблицю.

Авторами проаналізовано декілька конструкторів баз даних, в яких реалізовано візуалізацію зв'язків між таблицями: SchemaSpy [3], Database Designer [4], dbForge [5]. Встановлено, що переважно конструктори баз даних для візуалізації схеми бази даних у вигляді графа використовують алгоритм розкладки графів LinLog [6] та алгоритм центральних сил. Всі ці алгоритми ґрунтуються на підході фізичних аналогій [7].

4. ОПИС МОДИФІКОВАНОГО АЛГОРИТМУ

Авторами для підвищення якості візуалізації запропоновано модифікований двоетапний алгоритм з перетворенням координат.

Двоетапний алгоритм заснований на розділенні процесу розкладки графа на два етапу – грубу і тонку розкладку. На першому етапі алгоритмом типу центральних сил проводиться розкладка основи графа, що складається з вершин без підписів і ребер, кожна вершина апроксимується колом. Радіус кола визначається як сума радіусу вершини та діаметру кола, що охоплює підпис. У даній роботі для розкладки графа на цьому етапі використовується алгоритм LinLog [6]. Після закінчення першого етапу розкладки виконується зворотне перетворення графа – до отриманого на першому етапі графу додаються видалені підписи. При переході до другого етапу алгоритму виконується перетворення координат. Суть перетворення полягає в обчисленні коефіцієнту розтягування координат: для кожної пари розраховується найменша відстань, на якій гарантовано не буде накладення вершин і підписів, розрахована відстань ділиться на відстань між вершинами, отриману в результаті першого етапу кроку. Отриманий таким чином коефіцієнт розтягування використовується для перерахунку отриманих на першому кроці координат вершин. Після перетворення координат на другому етапі проводиться розкладка за допомогою розширеної енергетичної моделі. Розширена енергетична модель передбачає, що вершини взаємодіють відповідно до базової моделі LinLog [6], сил тяжіння між підписом до вершини та іншими вершинами немає, підпис взаємодіє лише з вершиною, якій він належить.

5. ЕКСПЕРИМЕНТАЛЬНА ОЦІНКА ЕФЕКТИВНОСТІ ВІЗУАЛІЗАЦІЇ ЗА МОДИФІКОВАНИМ АЛГОРИТМОМ

Вхідними даними для тестування ефективності візуалізації є певна база даних, що складається, як правило, з 10-400 таблиць. Перелік використаних для тестування баз даних наведено у таблиці 1.

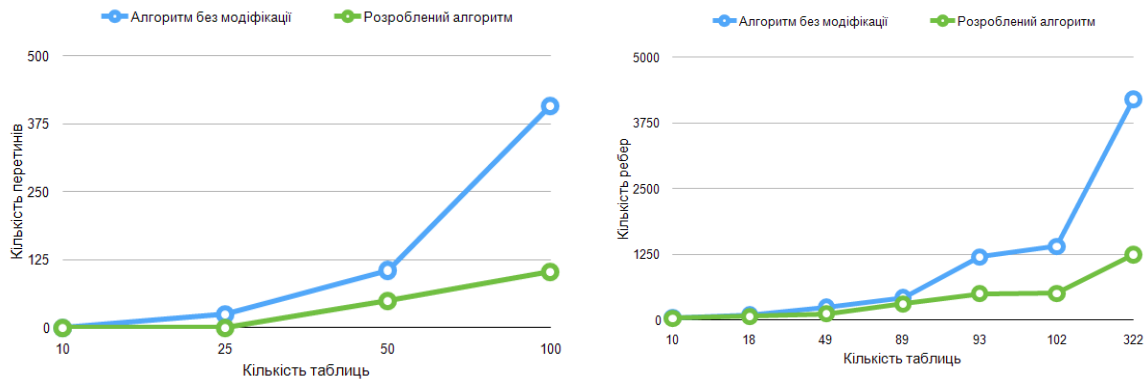
При візуалізації схеми кожної бази даних досліджувались естетичні критерії: мінімізація загального числа перетинів ребер, мінімізація областей, мінімізація ребер, мінімізація довжини всіх ребер.

На рисунку 2, а) показано кількість перетинів ліній зв'язків в базах даних з різною кількістю таблиць при візуалізації за допомогою алгоритмів з модифікацією і без неї. На рисунку 2, б) показано, як зростає кількість ребер в базах даних з різною кількістю таблиць при візуалізації за допомогою алгоритмів з модифікацією і без неї.

Таблиця 1 – Характеристики тестових баз даних

Назва	Кількість таблиць	Кількість зв'язків	Тип СУБД
Custom DB	10	32	MySQL
ReservationSpot	322	1243	MySQL
RadioSD	93	503	MSSQL
NewsPlace	49	120	MySQL
GuideMe	18	69	MySQL
JOJO Music	102	509	Postgre
MapGIS	89	308	Postgre

З рисунків видно, що модифікований алгоритм має переваги, і вони тим суттєвіші, чим більша база даних.



а) б)

Рисунок 2 – Характеристики алгоритмів для різної кількості таблиць

В результаті експериментів отримано, що двоетапні алгоритми з розтягуванням і без нього незначно відрізняються продуктивністю, але зображення, отримані двоетапним алгоритмом з розтягуванням, більш привабливі і задовольняють більшій кількості пред'явлених естетичних критеріїв розкладки, яки були розглянуті раніше.

6. ВИСНОВКИ

В роботі було виконано порівняння запропонованого авторами модифікованого алгоритму візуалізації графів з існуючими рішеннями,

які використовуються на реальних проектах. Порівняння виконувалось на основі естетичних критеріїв мінімізації: загального числа перетинів ребер, розмірів областей, загальної кількості ребер та загальної довжини ребер. Порівняння показало, що для алгоритму, розробленого авторами, критерії мінімізації розміру області та загальної кількості ребер також виконуються набагато краще, ніж у існуючих аналогів. Розроблений авторами алгоритм поступається аналогам за критерієм мінімізації загальної довжини ребер – це пояснюється особливістю алгоритму, яка передбачає автоматичне заповнення області зображення графа шляхом збільшення довжини ребер.

Дану розробку необхідно використовувати для тих проектів, які мають проблеми при візуалізації схеми бази даних. Це дозволить знайти вузькі місця в проектуванні бази даних і дозволить наглядно візуалізувати схеми, для подальшого вивчення та поліпшення.

ЛІТЕРАТУРА:

[1] Оре О. Теория графов – М. : Наука, 1980. – 340 с.
[2] Схема бази даних[Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/%D0%A1%D1%85%D0%B5%D0%BC%D0%B0_%D0%B1%D0%B0%D0%B7%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85
[3] SchemaSpy. Graphical Database Schema Metadata Browser [Електронний ресурс]. – Режим доступу: http://schemaspy.sourceforge.net/
[4] DbDesigner.net - Online database schema designer [Електронний ресурс]. – Режим доступу: https://dbdesigner.net/
[5] Основы работы с DbForge Studio - инструментом для работы с MySQL [Електронний ресурс]. – Режим доступу: http://fetisovvs.blogspot.com/2014/11/ dbforge-studio-mysql.html.
[6] Модель энергии LinLog. [Електронний ресурс]. – Режим доступу: http://mmc2.nsu.ru/default.aspx?db=book_apanovich2 &int=VIEW &el=1733&templ=I206
[7] Пупырев С.Н., Тихонов А.В. Визуализация динамических графов для анализа сложных сетей. Моделирование и анализ информационных систем. Т. 17, № 1 (2010), с. 117–135.

Отримано 21.06.2017

УДК 004.274

АНАЛІЗ РИНКУ КОМП'ЮТЕРНИХ ІГОР ЗАСОБАМИ ОН-ЛАЙН СИСТЕМ АНАЛІТИКИ

Н.С.Костюкова, І.А.Назарова, Є.В. Роговець
ДВНЗ «Донецький національний технічний університет»,
м.Покровськ, Україна,
E-mail: natalia.kostiukova@donntu.edu.ua

Робота присвячена огляду підходів до аналізу ринку програмного забезпечення з метою виявлення бажаних характеристик, які могли б забезпечити успішне просування програмного продукту на ринку. В статті розглядається процес аналізу на прикладі комп'ютерних ігор-головоломок. Показано, що для аналізу успішності розробленого програмного продукту можна використовувати системи аналітики, які звичайно пропонуються сервісами. Для вивчення ринку програм-аналогів перед початком розробки програми варто вивчити показники аналогів за допомогою характеристик, що надаються безкоштовними веб-сервісами аналітики.

Ключові слова: ключовий показник ефективності, веб- система аналітики, мобільний ігровий додаток-головоломка.

1. ВСТУП.

Починаючи розробку нового програмного продукту, зокрема, комп'ютерної гри, необхідно мати розуміння, чи потрібно це користувачам. За даними, наведеними в [1], кожного місяця в американському App Store виходить 12 тисяч нових ігор (тобто більше 400 ігор в день лише для однієї платформи). Така серйозна конкуренція істотно ускладнює просування нової гри на ринку. Починаючи розробку програмного продукту, слід розуміти, якими властивості він повинен мати, щоб стати успішним на ринку, які функції в ньому повинні бути реалізовані. Для цього варто вивчити програми- аналоги, з'ясувати, які з них більш успішні, хто є типовим користувачем і за які можливості цей користувач готовий платити. Після виходу продукту на ринок дуже важливо відслідковувати, наскільки активний на нього попит, чи є ефект від оновлень, що виходять. Для отримання цієї інформації потрібно проаналізувати ринок, отримати відповіді на поставлені запитання. Засобом, що дозволяє виконати такий аналіз, можуть стати системи аналітики.

Метою даної статті є аналіз можливостей веб-систем аналітики, що дозволяють оцінювати програмний продукт як на етапі розробки, так і на стадії підтримки вже розробленого продукту (випуск петчів і оновлень),

виділення показників, які дозволяють оцінити додатки класу «мобільний ігрове додаток-головоломка».

Об'єктом дослідження в даній статті є мобільні ігрові програми-головоломки. Предметом дослідження є оцінка характеристик таких додатків при їх появі на ринку з метою успішного просування.

Дослідження виконані в рамках проекту Erasmus + №2015-3181 / 001-003 "GameHub: University-enterprises cooperation in game industry in Ukraine".

2. ІНСТРУМЕНТИ АНАЛІЗУ РИНКУ МОБІЛЬНИХ КОМП'ЮТЕРНИХ ІГОР.

Рекомендації для успішного просування гри, наведені в [2], включають правильний вибір цільової аудиторії, підготовленість до будь-яких технічних проблем, активний супровід випущеної версії, своєчасну ліквідацію виявлених при експлуатації проблем, збереження вдалих рішень при випуску нових версій.

Якщо версія гри вже випущена і мова йде про випуск петчів та оновлень, розробник може скористатися доступними сервісами - системами аналітики. Інструменти, реалізовані в системах аналітики, включають в себе розрахунок метрик, що дозволяють оцінити ризик від розробки, і точні цілі, яких потрібно досягти. Ці метрики в сукупності називаються ключовими показниками ефективності (англ. Key Performance Indicators, або KPIs) [8, 9, 10]. Другий важливий інструмент систем аналітики - когортний аналіз, який передбачає виділення і оцінку певної групи користувачів (когорти) за певною ознакою [11, 12, 13].

3. АНАЛІЗ МОБІЛЬНИХ КОМП'ЮТЕРНИХ ІГОР-ГОЛОВОЛОМОК.

Сервіси (системи аналітики) для обчислення ключових показників ефективності можуть бути як зовнішніми, так і вбудованими в SDK проекту. Серед найбільш популярних можна назвати Flurry [3], Google Analytics [4], Mixpanel [5], Localytics [6], Game Analytics [7] та інші. Зазначені сервіси реалізують як обчислення KPI, так і когортний аналіз, відрізняються різноманітністю реалізованих функцій і зручністю роботи з ними. Серед безкоштовних сервісів, що дозволяють отримати показники програм, представлених на ринку, можна виділити AppAnnie [19], SensorTower [20], SimilarWeb [21].

На етапі проектування комп'ютерної гри, при аналізі ринку комп'ютерних ігор, що передує початку розробки, використовувати KPI для оцінки перспектив гри з певними характеристиками немає можливості, оскільки сервіси для обчислення KPI потрібно попередньо вбудовувати в додаток. Однак для такого аналізу можна використовувати інші

характеристики, доступні користувачам магазинів додатків та ігор, наприклад, Google Play. Такі метрики частково дозволяють охарактеризувати користувачів і рівень успіху представлених програм.

У даній статті аналізуються характеристики п'яти комп'ютерних однокористувальницьких ігор-головоломок, представлених в Google Play, які є конкурентами. Гра Kubik [14] випущена 20.12.2016. Гра Roll the Ball ™ - slide puzzle [15] вийшла 2.05.2015, останнє оновлення вийшло 11.04.2017. Гра Brain It On! - Physics Puzzles [16] вийшла 25.06.2016, дата виходу останнього оновлення 18.03.2017. Додаток Unblock me Free [17] випущено 16.05.2009, дата виходу останнього оновлення 8.12.2016. Гру Flow Free [18] випущено 7.06.2012, останнє оновлення вийшло 20.02.2017. Даний аналіз виконується з метою сформулювати характеристики гри-головоломки, яка, ймовірно, буде успішною.

Перша з характеристик - Total Installs - показник всіх установок програми з магазину. Така метрика збирається безпосередньо з магазину і характеризує поширеність додатку.

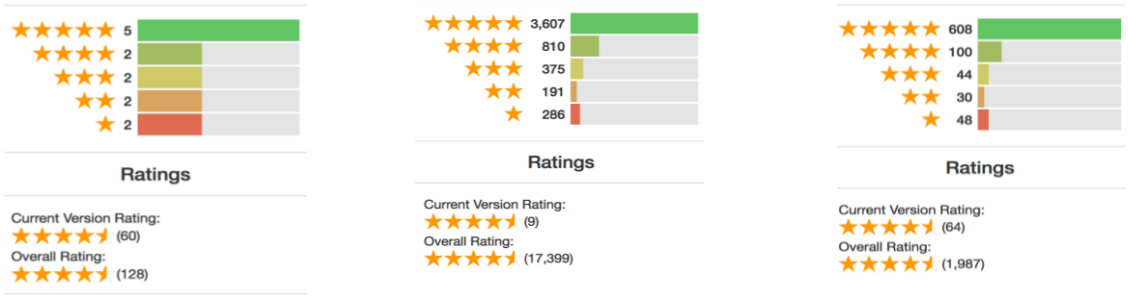
Наступна характеристика, доступна для ігор, пропонованих в Google Play - Rating by ... - суб'єктивна оцінка додатку. Це оцінка за шкалою від 1 до 5, що виставляється за відгуками і оцінкою проекту в магазині, причому є можливість подивитися оцінки як поточної версії, так і всіх версій, у всьому світі або по окремим країнам. На рис. 1, а)-д) наведені рейтинги розглянутих ігор (за даними AppAnnie). Як видно з малюнка, всі ігри мають приблизно рівний рейтинг, тому вони придатні для спільного аналізу.

Уважне вивчення даних показників дозволить побачити інтерес користувачів певної країни до програми в цілому і її версій (зокрема, зниження рейтингу гри в якійсь державі після випуску чергового оновлення говорить про помилковість прийнятого рішення).

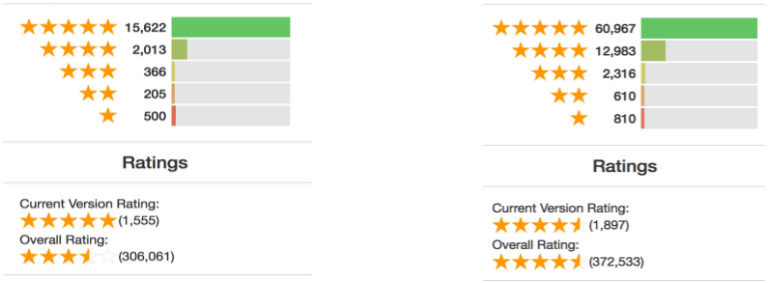
Ще одна характеристика - Review Breakdown Per Day- показує кількість відгуків в день. Значення характеристики для розглянутих ігор наведені на рис. 2, а)-д) (за даними SensorTower).

Характеристика From around the web to the store page - показник ефективності реклами. Ця характеристика дозволяє дізнатися, звідки користувачі дізнаються про програму і переходять на скачування, і відповідно, зробити висновок про доцільність того чи іншого способу реклами. На рис. 3 показані значення даної характеристики для аналізованих комп'ютерних ігор (SimilarWeb).

Метрика From Play Store to the store показує ефективність ключових слів при пошуку програми в пошукових системах і магазинах (див.рис.4, за даними SimilarWeb). Аналіз даного показника дозволить вибрати набір ключових слів, які сприяють збільшенню кількості знаходженню посилань на додаток



а) б) в)



г) д)

Рисунок 1 – Рейтинги ігор:
а) Kubik, б) Roll the Ball™ - slide puzzle, в) Brain It On! - Physics Puzzles, г) Unblock me Free, д) Flow Free



а) б)

в) г)

д)

Рисунок 2 - Активність за відгуками:
а) Kubik, б) Roll the Ball™ - slide puzzle, в) Brain It On! - Physics Puzzles, г) Unblock me Free, д) Flow Free

.

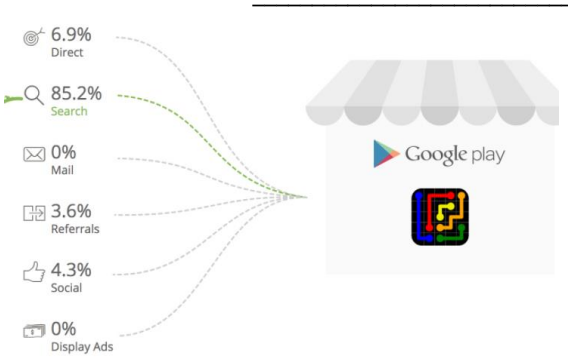


Рисунок 3 - Джерела реклами гри Flow Free

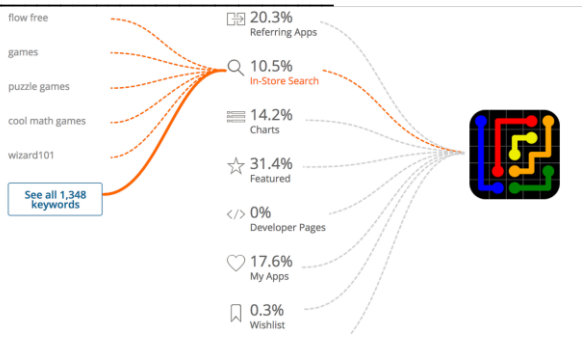


Рисунок 4 – Пошукові запити гри Flow Free

В цілому показники ефективності реклами (за даними SimilarWeb) наведені на рисунку 5, де показана частка різних джерел реклами (у відсотках). Як видно, найменш ефективна реклама по електронній пошті, найбільш ефективна – та, що розміщується на сторінках пошуку. Реклама в соціальних мережах дає ефект через певний проміжок часу (наприклад, для додатка Unblock me Free - найстарішого з розглянутих - вона ефективна майже також, як і реклама на сторінках пошуку).

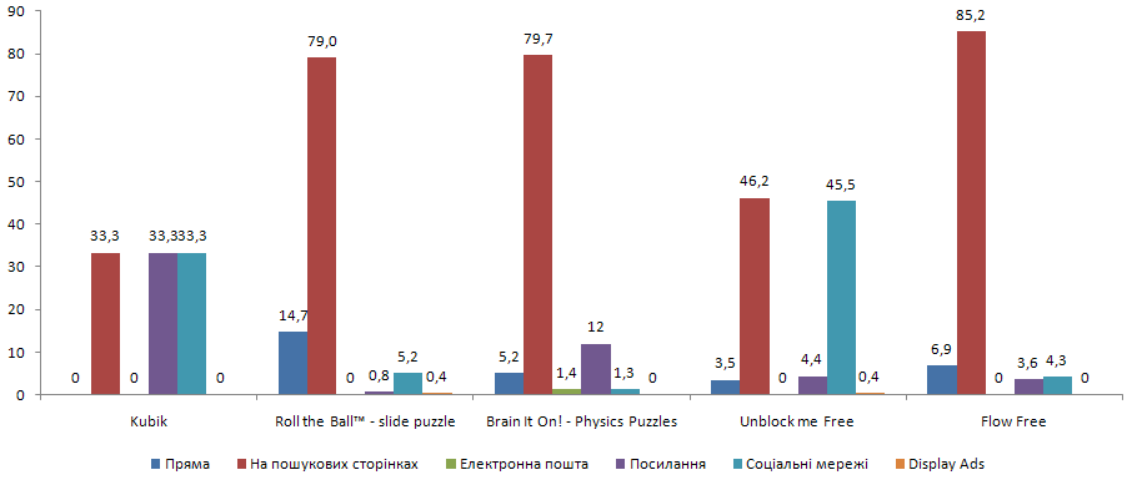
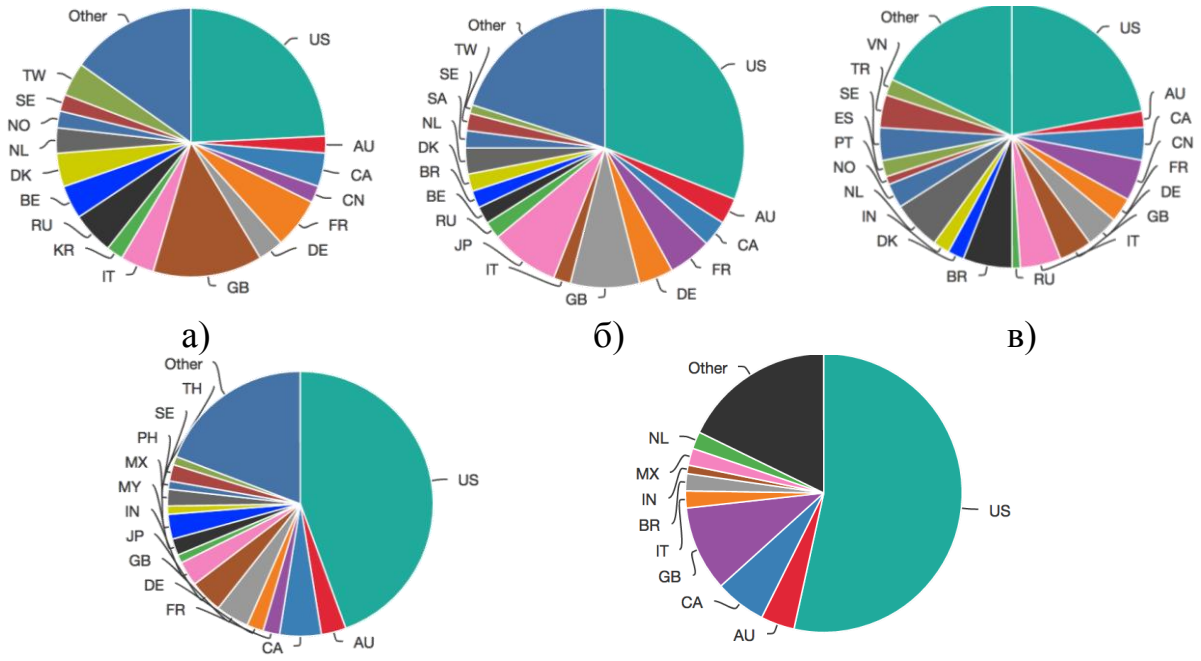


Рисунок 5 - Джерела реклами ігор (доля у відсотках)

Можливість аналізувати країни користувачів (User by Country) може бути корисною при локалізації програмного продукту: наприклад, якщо гра випущена з англомовним інтерфейсом, її рейтинг в певній країні високий, а кількість користувачів невелика, є сенс випустити версію на мові, поширеній в даній країні. На рис.6 показано розподіл користувачів ігрових додатків за їх країнами (за даними SimilarWeb).

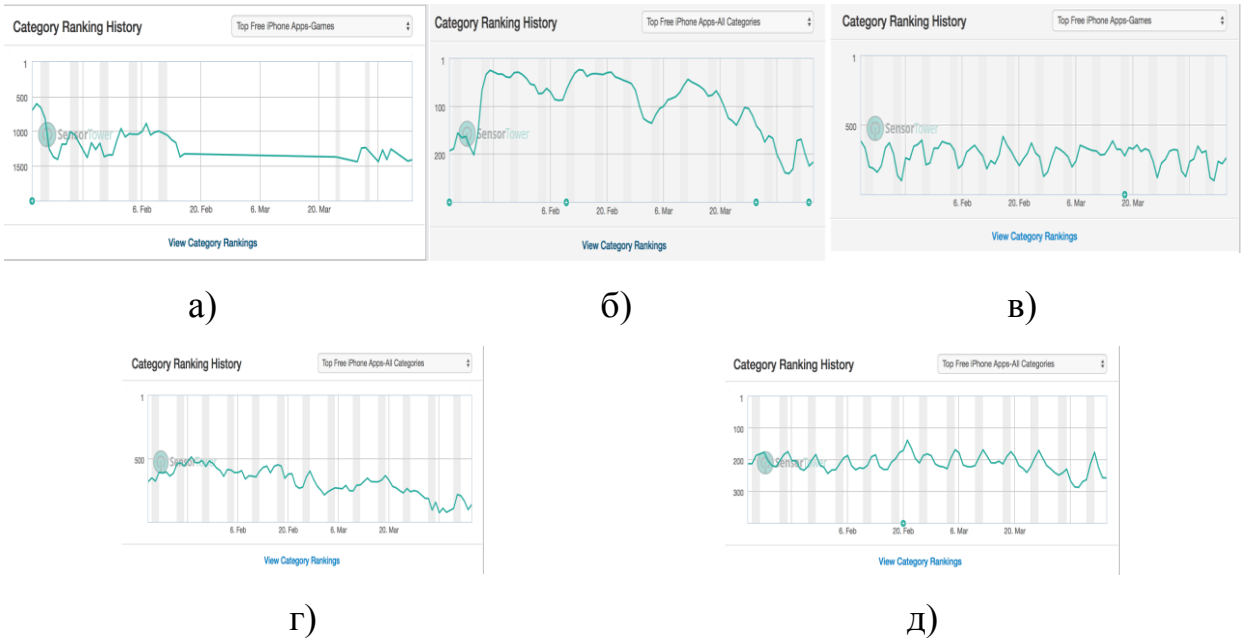
Наступна характеристика, яка вивчалася для розглянутих ігор - Category ranked - показує, як змінюється рейтинг гри в рамках свого жанру, дозволяючи зробити висновки про те, що позитивно сприймається аудиторією, а що змушує рейтинг знижуватися (див.рис.7, за даними SensorTower).



а) б) в)

г) д)

Рисунок 6 – Розподіл користувачів ігор за країнами:
а) Kubik, б) Roll the Ball™ - slide puzzle, в) Brain It On! - Physics Puzzles, г) Unblock me Free, д) Flow Free



а) б) в)

г) д)

Рисунок 7 – Статистика рейтингу ігор:
а) Kubik, б) Roll the Ball™ - slide puzzle, в) Brain It On! - Physics Puzzles, г) Unblock me Free, д) Flow Free

Цікавим є аналіз даної характеристики з урахуванням статистики оновлень. Так, падіння рейтингу після виходу чергового оновлення

свідчить про те, що оновлення було невдалим і необхідно виправляти ситуацію, а зростання рейтингу в аналогічній ситуації є приводом почекати з випуском оновлень.

Ще однією важливою можливістю систем аналітики є статистика про пристрої, на яких функціонує додаток. Рідке використання деяких пристроїв може свідчити про проблеми з експлуатацією на них програми, що дуже важливо для розробників. Ця інформація була б корисна і при проектуванні нового додатка, даючи уявлення про незаповнені ніші на ринку програмного забезпечення. Однак такі дані доступні тільки для особистого кабінету Google Analytics конкретного додатку.

4. **ВИСНОВКИ**.

В роботі виконано огляд підходів до аналізу ринку програмного забезпечення з метою виявлення бажаних характеристик, які могли б забезпечити успішне просування програмного продукту на ринку і розглянутий процес аналізу на прикладі комп'ютерних ігор-головоломок. За результатами огляду можна зробити наступні висновки: вивчення показників продажів в магазинах додатків та ігор дозволяє встановити, чи успішним було виконане оновлення програми, чи є проблеми з його роботою на різних типах пристроїв, чи є перешкоди для його поширення в певному регіоні.

Перспективним напрямком подальшої роботи є виділення набору показників, що дозволяють оцінити бажані характеристики комп'ютерної гри кожного жанру, на підставі якого можна прийняти обґрунтоване рішення про розробку.

ЛІТЕРАТУРА

[1] **Шишкова О.** KPI в играх: Какие метрики необходимо измерять игровому дизайнеру [Електронний ресурс]: *https://vc.ru/p/games-kpi*

[2] **Rogers S.** Level Up! The Guide to Great Video Game Design, 2nd Edition. ISBN: 978-1-118-87716-6. *Wiley, New York,* 550 p.

[3] Yahoo mobile developer suite for you apps: заголовок з екрану. *[Електронний ресурс]: https://developer.yahoo.com/*

[4] Веб-аналитика корпоративного уровня: заголовок з екрану. *[Електронний ресурс]: https://www.google.ru/analytics/*

[5] Product analytics for product people: заголовок з екрану. *[Електронний ресурс]: https://mixpanel.com/*

[6] Engage your mobile users how they want, when they want: заголовок з екрану. *[Електронний ресурс]: https://www.localytics.com/*

[7] To build great games, you need to understand player behavior: заголовок з екрану. *[Електронний ресурс]: http://www.gameanalytics.com/*

[8] **Семенов А**. Зачем в играх нужна предиктивная аналитика?: *[Електронний ресурс]: http://app2top.ru/marketing/zachem-v-igrah-nuzhna-prediktivnaya-analitika-96554.html*
[9] Аналитика игр - удержание: заголовок з екрану. *[Електронний ресурс]: http://alexddd.com/game-metrics-retention/*
[9] **Сахнов К.** Аналитика для геймдизайнеров и продюсеров. Часть I *[Електронний ресурс]: https://habrahabr.ru/company/miip/blog/309444/*
[10] Alawar Entertainment. Анализ статистики во free-to-play играх: инструменты аналитика. *[Електронний ресурс]: https://habrahabr.ru/company/alawar/blog/162739*
[11] **Очкова Е.** Как выглядит идеальная система аналитики для игрового проекта. *[Електронний ресурс]: https://vc.ru/p/game-analytics-2*
[12] **Siegel E.** Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die, *Wiley, New York*, 2016, 368 p.
[13] Kubik *[Електронний ресурс]: https://play.google.com/store/apps/details?id=com.ketchapp.kubik*
[14] Roll the ball *[Електронний ресурс]: https://play.google.com/store/apps/details? id=com.bitmango.rolltheballunrollme&hl=ru*
[15] Brain It On! - Physics Puzzles *[Електронний ресурс]: https://play.google.com/store/apps/details? id=com.orbital.brainiton&hl=ru*
[16] Unblock me free *[Електронний ресурс]: https://play.google.com/store/apps/details? id=com.kiragames.unblockmefree&hl=ru*
[17] Flow Free *[Електронний ресурс]: https://play.google.com/store/apps/details? id=com.bigduckgames.flow&hl=ru*
[18] AppAnnie system *[Електронний ресурс]: https://www.appannie.com*
[19] SensorTower system *[Електронний ресурс]: https://sensortower.com*
[20] SimilarWeb System *[Електронний ресурс]: https://www.similarweb.com*

Отримано 21.06.2017

УДК004.942

АНАЛІЗ МОДЕЛІ ТА АЛГОРИТМІВ, ЩО МОЖУТЬ БУТИ ВИКОРИСТАНІ ДЛЯ ПОБУДОВИ ІГРОВОГО ФІЗИЧНОГО РУШІЯ ДЛЯ ТВЕРДИХ ЧАСТОК

М.М.Лефтеров
Донецький національний технічний університет

У роботі розглядаються основі поняття «Фізичний рушій», «Частка», «Просторова база даних». Розглянуті основні фізичні закони для симуляції кінематики твердих часток, існуючі ігрові фізичні рушії та алгоритми, які можуть бути використані у фізичному рушії.

Ключові слова: фізичний рушій, ігровий фізичний рушій, частка, чисельне інтегрування, просторова база даних.

1. ВСТУП

Багато сучасних ігор симулюють дії та процеси з реального життя (наприклад, рух автомобіля, вільне падіння крапель дощу та ін.) і чим краще буде візуально відтворено це, тим більше людей буде хотіти пограти у ці ігри, а компанія розробник отримає більше грошей. Для цього іграм необхідно мати такий компонент, як фізичний рушій.

Актуальність полягає у тому, що симуляція фізики є складною задачею і комп'ютери типових користувачів не можуть у повній мірі виконувати цю симуляцію на великих масивах даних, тому треба розробити алгоритм, що виконується найшвидше при використанні модифікацій та комбінацій існуючих алгоритмів.

2. ПОСТАНОВКА ЗАДАЧІ

Об'єктом дослідження в роботі є математичні методи для фізичної симуляції.

Предметом дослідження є математична модель кінематики системи твердих часток.

Метою дослідження є ігровий фізичний рушій для симуляції твердих часток, що має найбільшу швидкість виконання серед можливих модифікацій та комбінацій існуючих алгоритмів.

Для реалізації поставленої мети в роботі вирішуються наступні завдання:

1) пошук алгоритмів, які можливо використати у фізичному рушії;

2) проектування та розробка програмної системи-ігрового фізичного рушія;

3) проведення тестування різних алгоритмів та різних умов виконання симуляції.

3. ОГЛЯД ТЕМИ

Фізичний рушій – комп'ютерна програма, що робить симуляцію фізичних законів реального світу у світі віртуальному з тим або іншим ступенем апроксимації.

Ігровий фізичний рушій – фізичний рушій, що використовується в комп'ютерних іграх як компонент ігрового рушія. У цьому випадку він повинен працювати в режимі реального часу, тобто відтворювати фізичні процеси в грі з тою ж швидкістю, з якою вони відбуваються в реальному світі. Разом із тим, від ігрового фізичного рушія не потрібно точності обчислень. Головна вимога – візуальна реалістичність, – і для її досягнення не обов'язково проводити точну симуляцію. Тому в іграх використовуються дуже приблизні апроксимації, наближені моделі й інші програмні «трюки».

Частка – об'єкт фізичної симуляції, що має масу.

Тверда частка – частка, що має радіус взаємодії з іншими частками.

Просторова база даних – база даних, що оптимізована для зберігання і виконання запитів до даних про просторові об'єкти, представлених деякими абстракціями: точка, лінія, полігон та ін. Просторові БД дозволяють виконувати аналітичні запити, що містять просторові оператори для аналізу просторово-логічних відносин об'єктів (перетинається, стосується, міститься в, містить, знаходиться на відстані X від, збігається та ін.) [1].

Фізичний рушій має виконувати симуляцію деяких фізичних процесів. Такими процесами є: рух, прискорення, ривок, взаємодія тіл, імпульс.

Рух – зміна положення тіла на наступному такті часу виходячи з даних поточного такту. Цей процес має використовувати метод чисельного інтегрування швидкості по часу.

Прискорення – зміна швидкості тіла на наступному такті часу виходячи з даних поточного такту. Цей процес має використовувати метод чисельного інтегрування прискорення по часу (формула 3.1), принцип суперпозиції відносно сил, діючих на тіло (формула 3.2) та другий закон Ньютона (формула 3.3).

$$\vec{a} = \frac{d\vec{v}}{dt} \tag{3.1}$$

$$\vec{F} = \sum_{i=1}^{n} \vec{F_i} \tag{3.2}$$

$$\vec{F} = m\vec{a} \tag{3.3}$$

де:

$\vec{a}$ – прискорення тіла;

$\vec{F}$ – результуюча сила, діюча на тіло (суперпозиція сил);

$\vec{F_i}$ – окрема сила, діюча на тіло;

n – кількість окремих сил;

m – маса тіла.

Ривок – зміна прискорення тіла на наступному такті часу виходячи з даних поточного такту. Цей процес має використовувати метод чисельного інтегрування ривка по часу (формула 3.4). Ривок може використовуватися у деяких методах чисельного інтегрування.

$$\vec{j} = \frac{d\vec{a}}{dt} \tag{3.4}$$

де:

$\vec{j}$ – ривок.

Взаємодія тіл – те, як тіла діють одне на одне. Стандартно вони бувають таких типів: зіткнення та силові взаємодії тіл, а фізичний рушій додатково може надавати можливість визначати власні правила взаємодії та з'єднання тіл.

Під час **зіткнення** використовується закон збереження імпульсу та просторова база даних. Просторова база даних у випадку фізичного рушія має вирішувати задачі взаємного розташування тіл та видавати всі випадки перетину меж тіл.

Після зіткнення фізичний рушій повинен роздвинути тіла, щоб їх межі не перетиналися.

Під час **силової взаємодії** використовується третій закон Ньютона (формула 3.5) та функція, що визначає закон взаємодії, використовуючи параметри тіл (формула 3.6):

$$\vec{F}_{1\to 2} = -\vec{F}_{2\to 1} \tag{3.5}$$

$$\vec{F_i} = f(x_i, y_i, \vec{v_i}, m_i, \dots) \tag{3.6}$$

де:

$\vec{F}_{1\to2}$ – сила дії першого тіла на друге;
$\vec{F}_{2\to1}$ – сила дії другого тіла на перше;
f – функція, що визначає закон взаємодії цих тіл.

Імпульс – моментальна та одноразова зміна швидкості тіла. Може виникати через якісь специфічні події. Наприклад, під час початкового запуску тіла. Імпульс використовує другий закон Ньютона (формула 3.7) та власне визначення (формула 3.8).

$$\vec{F} = \frac{d\vec{p}}{dt} \tag{3.7}$$

$$\vec{p} = m\vec{v} \tag{3.8}$$

де:
$\vec{p}$ – імпульс тіла.

4. ОГЛЯД АЛГОРИТМІВ ТА СТРУКТУР ДАНИХ

Деякі дані та обмеження у фізичному рушії, до яких будуть використовуватися алгоритми інтегрування:

– час між двома тактами фіксований, тобто $dt = const$;

– у якості змінної інтегрування виступають координати тіл, проекції їх швидкостей та прискореннь на координатні вісі.

Метод Ейлера – найпростіший чисельний метод рішення систем звичайних диференціальних рівнянь. Метод Ейлера є явним, однокроковим методом першого порядку точності, заснованому на апроксимації інтегральної кривої кусочно-лінійною функцією, так званої ламаної Ейлера [2]. Цей метод має похибку на кроці $O(h^2)$ і похибка в цілому $O(h)$.

Методи Рунге-Кутта – великий клас чисельних методів розв'язання задачі Коші для звичайних диференціальних рівнянь і їх систем.

Метод Рунге-Кутта четвертого порядку при обчисленнях із постійним кроком інтегрування настільки широко поширений, що його часто називають просто методом Рунге-Кутта. Цей метод має четвертий порядок точності. Це означає, що помилка на одному кроці має порядок $O(h^5)$, а сумарна помилка на кінцевому інтервалі інтегрування має порядок $O(h^4)$.

Чисельний **метод** інтегрування **Верле** – чисельний метод, який використовується для інтегрування рівнянь руху матеріальної точки ($\ddot{\vec{r}} = \vec{a}(\vec{r}, t)$). Часто використовується для обчислення траєкторій

частинок в моделях молекулярної динаміки і в комп'ютерних іграх. Метод Верле більш стійкий, ніж більш простий метод Ейлера, і має при цьому інші якості, необхідні для моделювання фізичних процесів в реальному часі [3]. Алгоритм Верле використовується для обчислення наступного місця розташування точки з поточного та минулого, без використання швидкості.

Основна особливість алгоритму полягає в можливості накладати на систему точок різні обмеження. Наприклад, можна зв'язати деякі з них твердими стрижнями заданої довжини.

QuadTree (дерево квадрантів, квадродерево, 4-дерево) – дерево, в якому у кожного внутрішнього вузла рівно 4 нащадка. Дерева квадрантів часто використовуються для рекурсивного розбиття двомірного простору по 4 квадранта (області). Областями є квадрати, прямокутники або мають довільну форму [4].

AABB (Axis-Aligned Bounding Box) – це обмежена область в просторі у вигляді прямокутного паралелепіпеда, зі сторонами, паралельними осям координат у світовій системі. При обертанні об'єкта AABB змінює свої розміри, але завжди залишається орієнтованим по осях координат. Може використовуватися у якості даних про тіло у структурі QuadTree.

5. ВИСНОВКИ

В ході аналізу були виявлені типові процеси, які має симулювати фізичний рушій, а саме: рух, прискорення, ривок, взаємодія тіл, імпульс.

Було розглянуто наступні алгоритми та структури даних: метод Ейлера, методи Рунге-Кутта, метод Верле, структура QuadTree.

ЛІТЕРАТУРА:

1. Основы пространственных баз данных / Шекхар Шаши, Чаула Санжей. — Москва: Кудиц-образ, 2004. — 336 с.

2. Эйлер Л. Интегральное исчисление, том 1 / пер. з лат. С.Я.Лурье, М.Я.Выгодского // Интегральное исчисление. – 1956. – 415 с.

3. A molecular dynamics primer, Furio Ercolessi, International School for Advanced Studies, Trieste, Italy (SISSA-ISAS), pp 15-16, 1997.

4. Дерево квадрантов [Електронний ресурс]: «Wikipedia». – https://ru.wikipedia.org/wiki/Дерево_квадрантов – Дата доступу: вересень 2017. – Загол. з екрану.

Отримано 29.06.2017

УДК 004.051

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ЗАСТОСУВАННЯ ПРОЦЕДУРНОЇ ГЕНЕРАЦІЇ В КОМП'ЮТЕРНИХ ІГРАХ

Д. А. Прядка, Н.О. Маслова

Донецький національний технічний університет, Україна, Покровськ.

Надано визначення поняттю процедурної генерації, сформульована мета розробки, вказано досліджувані параметри якості, наведено перелік застосованих алгоритмів та їх особливості.

Ключові слова: процедурна генерація, лабіринт, характеристики якості, алгоритми генерації, пошук найкоротшого шляху.

Під процедурною генерацією розуміють автоматичне і напівавтоматичне створення і динамічну зміну різних складових частин ігор, в тому числі ігрових об'єктів і рівнів, двовимірної і тривимірної графіки, ефектів, звуків, музики, персонажів, сюжетів [1].

Необхідність використання процедурної генерації контенту полягає в мінімізації витрат на цю роботу при одночасному досягненні найкращої якості. У можливості генерувати контент в обсязі, який було б затратно або неможливо в принципі створити вручну.

У науковому середовищі інтерес до проблеми процедурної генерації контенту в відеоіграх з'явився досить недавно й в наш час є актуальним.

Мета роботи – опанувати методи автоматичної генерації, визначити переваги та недоліки різних підходів до створення грального контенту, отримати характеристики якості контенту, згенерованого різними методами та надати рекомендації щодо обрання найбільш ефективних за обраними параметрами методів та алгоритмів.

В якості базових обрані методи створення двовимірних лабіринтів. В якості основних характеристик при оцінці якості алгоритмів створення двовимірних лабіринтів виділяють параметри, які отримали назву упередженість, рівномірність, пам'ять і час [1].

Упередженість - це нерівна ймовірність побудови проходу між клітинами лабіринту в залежності від напрямку. Під рівномірністю розуміється рівноймовірність появи будь-якого з можливих результатів генерації.

Параметр пам'ять передбачає визначення обсягу пам'яті, необхідного для роботи алгоритму. Останній з обраних параметрів характеризує час виконання алгоритму.

У конкретному випадку розробки модуля, також слід враховувати простоту реалізації алгоритму на увазі його можливої подальшої модифікації.

В якості методів генерації двовимірних лабіринтів в роботі розглянуті два алгоритму: алгоритм Краскала і алгоритм Ейлера.

Алгоритм Краскала, в даному випадку, - це алгоритм побудови мінімального каркасного дерева, адаптований для побудови ідеального лабіринту [2].

Алгоритм Ейлера відноситься до типу, званому в англомовних джерелах Simple algorithms. Такі алгоритми зберігають в пам'яті тільки поточний і попередній рядок лабіринту, а також зв'язки між ними.

Алгоритм Краскала має найбільш просту і зрозумілу реалізацію. Разом з тим, алгоритм Ейлера характеризується найбільшою швидкістю генерації і економним використанням пам'яті, при трохи більшій упередженості і меншій рівномірності.

Будь який лабіринт передбачає можливість проходження. В якості алгоритму пошуку найкоротшого шляху обрано алгоритм Дейкстри [3].

Для реалізації поставлених завдань використовується потужне середовище розробки «Unity 3D» з використанням мови програмування C#. Робота алгоритмів демонструється за допомогою кадрів комп'ютерної гри з автоматичною генерацією полів лабіринтів.

Переваги процедурної генерації включають менші розміри файлів, більшу різноманітність контенту і випадковість для менш передбачуваного геймплея.

ПЕРЕЛІК ПОСИЛАНЬ:

[1] Togelius J. Procedural Content Generation in Games: A Textbook and an Overview of Current Research / J. Togelius N. Shaker M. Nelson Berlin: Springer, 2014, 142p.

[2] Kodicek, Danny Mathematics and physics for programmers /Danny Kodicek, John P. Flynt-Hingham, Mass.: Charles River Media, 2005, 541p.

[3] Алгоритм Дейкстри [Електронний ресурс] http://e-maxx.ru/algo/dijkstra, Режим доступу: вільний (дата доступу – 28.06.2017).

Отримано 29.07.2017

УДК 004.932.2

COMPUTERIZED SPERM QUALITY ANALYZER

Aleksandr I. Nikolskyy [a], Yosyp Y. Bilynskyy [a], Alexander A Lazarev [a], Vladimir G. Krasilenko[b]
[a] Vinnytsia National Technical University, Ukraine;
[b] Vinnytsia Social Economy Institute, Vinnytsia, Ukraine;

В роботі розглянуто використання системи Labview для автоматизації процесу аналізу якості сперми. За спермограмою ми визначаємо кількість та рухливість сперматозоїдів, а також помічаємо найкращих. Для вхідного зображення виконуємо попередню обробку для покращення якості зображення. Розроблена система передбачає обробку в реальному режимі часу з можливістю адаптації під розмір та якість спермограми.

1. INTRODUCTION

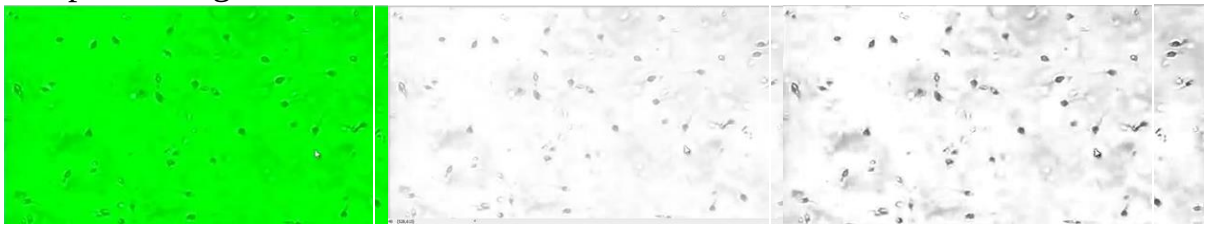
Modern technologies of computer processing and image analysis [1-3] can improve the quantitative and qualitative indices of the patient's spermogram analysis. The main problem of tracking biological objects in video stream is automatic finding them in each separate frame. When analyzing moving biological objects recorded by digital microscope using tracking algorithms, a large number of frames must be processed in the images stream. Counting the number of selected objects in each frame to determine the average of a standard set, usually 200 frames, and grouping and clustering by different parameters, including shape, speed, location, etc., are important subtasks [1]. And for each of these sub-tasks not only dedicated optimal image processing algorithms adapted to specific conditions, but the possibility to rearrange and integrate them easily into a single tool environment are required. There are well-established approaches to recognition of very noisy and correlated single objects [4, 5] and sets of multiple objects [6], including moving ones [7], with simultaneous division into clusters. However, they are all very diverse and poorly integrated into single, flexible and configurable, and adaptive system or program. Therefore, when choosing a tool for research, we settled on Labview as the most powerful system-design platform and development environment for a visual programming language by National Instruments. [8].

2. EXPERIMENTAL STUDIES OF IMAGE PRE-PROCESSING ALGORITHMS FOR EXTRACTION AND ANALYSIS OF MULTIPLE BIOLOGICAL OBJECTS

One of the main objectives of research of tracking algorithm of moving biological objects is pre-processing of frames in the stream to improve the quality of their subsequent detection and tracking.

2.1 The algorithm of frames pre-processing, using morphological tool and local threshold

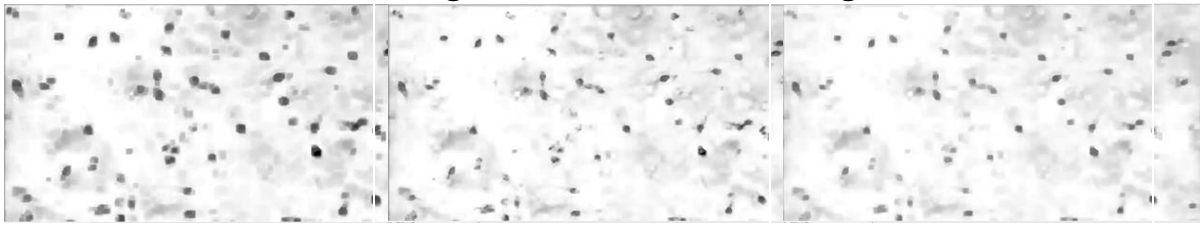
To eliminate the problems described above, one of pre-processing algorithms can be used (Fig.1). As shown in Figure 1a the original color image in addition to the objects of interest, has many false objects and complex background.



a) Original Image RGB (32 bits) 1189×619

b) Color Plan Extraction: Extracts the color plan (RGB – Green plan) from an image.

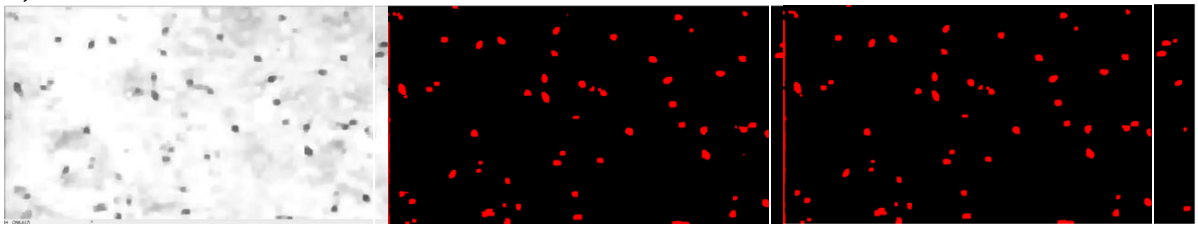
c) Brightness: Alters the brightness (85), contrast (53,7), and gamma (1,61) of an image.



d) Gray Morphology_1: Modifies the shape of objects in an image. (Operation: Erode – Structuring Element cross 3×3, Iterations - 4)

e) Gray Morphology 2: Modifies the shape of object in an image. (Operation: Dilate – Structuring Element square 3×3, Iterations - 3)

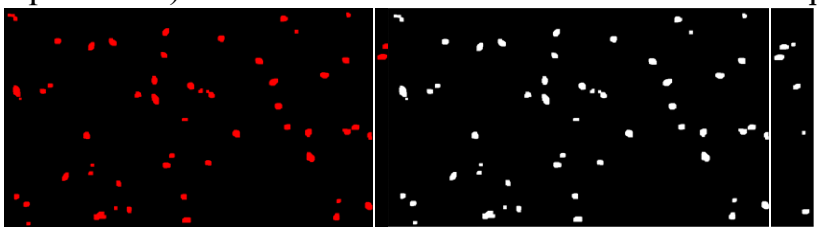
f) Gray Morphology_3: Modifies the shape of object in an images. (Operation: Close – Structuring Element square 9×9)



g) Gray Morphology_4: Modifies the shape of object in an image. (Operation: Open – Structuring Element square 9×9)

h) Threshold: Selects ranges of pixel values in grayscale in an image. (Local Threshold: Background Correction Look For: Dark Object)

i) Basic Morphology: Modifies the shape of binary objects in an image. (Operation: Auto Median – Structuring Element square 7×7)



j) Adv. Morphology: (Operation: Remove border objects – Structuring Element square 3×3 - Connectivity 4/8)

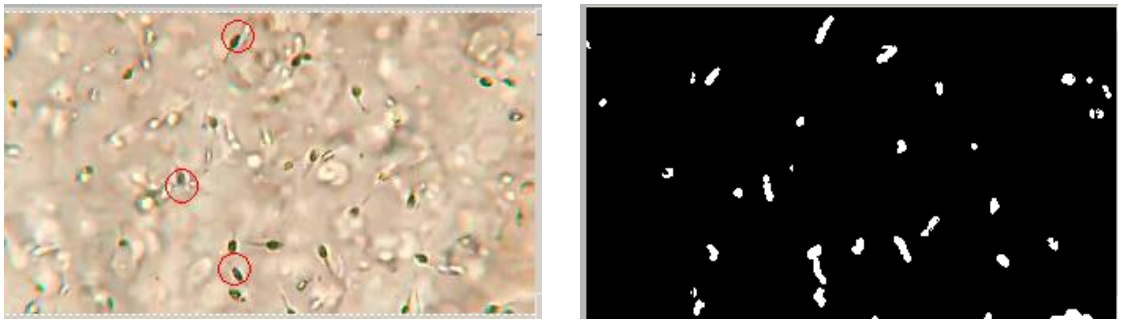
k) Color Operators: Performs arithmetic and logical operations on images. (Operation: Multiply 255)

Figure 1. The algorithm of frames pre-processing using a morphological tool and local threshold, version 1

Preprocessing includes equalization of common luminance of the image, elimination of the high frequency noise and various artifacts (highlighting of the separate regions, gaps, fractures, etc.) in the original image, contrast adjustments, threshold processing, binarization, and other functional transformations. Preprocessing by one of the algorithms presented in Fig.1 includes a set of tools described above, which enable successive transformation of each image frame.

3. USING LABVIEW FOR REAL-TIME SIMULATION, CLASSIFICATION, TRACKING AND MATCHING MULTIPLE BIOLOGICAL OBJECTS

The main indicators of sperm quality are the number of spermatozoons and their mobility. After preliminary image processing using NI Vision Assistant and Fuction Practicle Analysis, we will count the number of spermatozoa and determine their basic parameters. Fig. 2 shows the input frame, the frame after preprocessing and the determination of mobility.



a) Input frame with indicated best spems b) Frame after mobility analisys
Figure 2. Simulation results of spermatozoons analisys

In this example, the number of particles is 24, and the main parameters of the first five particles, namely the position and size, are shown in Table 1.

Table 1. Parameters of the first 5 particles after analysis

Object #	Center of Mass X	Center of Mass Y	Bounding Rect Diagonal	Perimeter	Max Feret Diameter	Area	% Area/Image Area
1	529,57426	24,40594	16,97056	37,18959	14,42221	101	0,04384
2	430,05806	27,90323	22,80351	56,25136	19,31321	155	0,06727
3	120,67442	49,81395	12,80625	27,70276	10,19804	43	0,01866
4	220,13274	64,9469	18,02776	39,61967	15,52417	113	0,04905
5	618,75281	66,94382	15,81139	35,53274	13,60147	89	0,03863

To determine the motility of the spermatozoons, we perform a logical sum operation on the frames. During the motion of the sperms, their trajectories will be formed, according to which we can determine the number of progressively fast moving (class A), the number of progressively slow mobility (class B), the number of low active (class C) and fixed (class D).

4.CONCLUSIONS

The tasks of real-time tracking of the dynamics of movement of various biological objects are researched in the work. Specificity of these objects, conditions of their visualization and model parameters strongly influence the choice of methods and algorithms, which are optimal for a specific task. Therefore, in this article, in order to automate the processes of adaptation algorithm of recognition - tracking, we suggest frame pre-processing algorithm using NI Labview tools and Vision Assistant. Preprocessing included equalization of general background luminance of the image, elimination of high-frequency noise and different artifacts (highlighted areas, gaps, fractures, etc.) from the original image, contrasting, thresholding, binarization and other functional transformations. For further reconstruction, the moving objects in the image are enhanced

by sharpening and processed by algorithms. The paper shows the possibility of building frames preprocessing algorithms in the stream for a number of biological objects. Counting the number of selected objects in each frame to determine the average of a standard set, usually 200 frames, and grouping and clustering by different parameters, including shape, speed, location, are important subtasks there that are considered in this paper. Modern computer processing and image analysis techniques will improve the quantity and quality analysis of the patient's spermogram.

REFERENCE

[1] WHO laboratory manual for the examination and processing of human semen. — 5-d edition. — World Health Organization, 2010. — ISBN 978 92 4 1547789.

[2] Gonal J. S., Kohir V. V. "Morphological Segmentation of the Brain Tumors by Using Image Processing and LabVIEW", International Journal of Electronics, Electrical and Computational System, Volume 4, Special Issue March 2015, p.334-341

[3] Razali Tomari, Jalil Lias, Rabiatuladawiah Musa and Wan Nurshazwani Wan Zakaria, "Development of red blood cell analysis system using NI Vision Builder AI", ARPN Journal of Engineering and Applied Sciences, Vol. 10, NO 19, October, 2015, p.8692-8698

[4] Vladimir G. Krasilenko, Aleksandr Nikolskyy, Alexandr V. Zaitsev, Victor M. Voloshin, "Optical pattern recognition algorithms on neural-logic equivalent models and demonstration of their prospects and possible implementations", Proceedings of SPIE Vol. 4387, pp. 247-260.

[5] Vladimir G. Krasilenko, A. E. Nikolsky, M. Y. Bilynska, A. L. Pastushenko, "Prospects of liqud crystal structures application in instrumental realizations of neural network matrix-tensor equivalental models (MTEM)", Proceedings of SPIE Vol. 4938, pp. 212-222.

[6] Vladimir G. Krasilenko, Aleksandr Nikolskyy, Yuriy A. Bozniak, "Recognition algorithms of multilevel images of multicharacter identification objects based on nonlinear equivalent metrics and analysis of experimental data", Proceedings of SPIE Vol. 4731, pp. 154-163.

[7] Vladimir G. Krasilenko, Aleksandr Nikolskyy, Alexander A. Lazarev, "Modeling optical pattern recognition algorithms for object tracking based on nonlinear equivalent models and subtraction of frames", Proceedings of SPIE Vol. 9813.

[8] http://www.ni.com/

Received 26.06.2017

УДК 004. 925.9

МЕТОД УЗАГАЛЬНЕНОГО КОНТУРНОГО ПРЕПАРУВАННЯ ДЛЯ ОБРОБЛЕННЯ БІОМЕДИЧНИХ ЗОБРАЖЕНЬ

Павлов С.В., Ганиш Н.В., Бурденюк І.І.,
Чернишова Т.А., Поплавська А.А.
Вінницький національний технічний університет

У статті представлено реалізацію методу Q-препарування. В основі запропонованого метода лежить ідея скорочення об'єму обчислень при представленні зображення у вигляді Q – спектра та їх спрощення при перед обробці корелюємих зображень по адаптивному методу узагальненого контурного препарування. Адаптивність методу заключається в настройці порогу препарування по параметрам просторового розподілення яскравості вхідного зображення. Скорочення об'єму обчислень заключається в кодуванні препаратів попереднього зображення і заміні операцій множення на операції порівняння і підсумовування.

Актуальність. Однією з основних задач при обробленні біомедичної інформації, отриманої за допомогою оптико-електронних кореляційно-екстремальних засобів (ОЕКЕЗ) є виділення контурних зображень, які, як привило, становлять найбільшу цікавість і є найбільш інформативними. Це стає необхідним під час вирішення задач розпізнавання біомедичних образів та аналізі сцен, оскільки контур складає геометричну характеристику об'єкта і його ознак. Під час виділення границь зображень найбільш розповсюдженими вважаються методи, які базуються на різного роду статистичних та вірогідностних моделях, квантуванням функцій яскравості по її аргументам і значенням. Проте, слід відзначити, що об'єктивність отриманих результатів достатньо мала, особливо в умовах накладання на зображення різного роду шумів, викликаних особливостями ОЕКЕЗ, середовищем розташування, геометричними, енергетичними характеристиками об'єкта та ін. Тому створення сучасних ОЕКЕЗ, а особливо систем ідентифікації та моніторингу різних вільно-орієнтованих об'єктів, вимагає розробку нових, більш адаптованих до сучасних показників точності і якості, інтелектуальних (комбінованих) алгоритмів сегментації зображень.

Метод. Зображення реальних об'єктів, отримані у цифрових системах, надзвичайно рідко мають цифрове представлення, у зв'язку з цим їх переведення до даного вигляду є обов'язковим у тому випадку, якщо зображення підлягатиме цифровій обробці,

збереженню та передачі. Дана процедура складається з двох етапів: заміна постійного сигналу дискретним (дискретизація), заміна безперервної множини значень яскравості множеною квантованих значень (квантування). Проблеми виділення границь які виникають на даному етапі оброблення зображень пов'язані переважно з апаратними особливостями ОЕКЕЗ і полягають у нечіткому відображенні об'єктів з високою відносною швидкістю і отриманні фальшивих контурів при невеликій кількості рівнів квантування. У додаток зображення, сформовані різними інформаційними ОЕКЕЗ, спотворюються під дією завад. Також У ролі завад можуть виступати і компоненти самого зображення, які не несуть корисної інформації при вирішенні даної задачі.

В основі запропонованого метода лежить ідея скорочення об'єму обчислень при представленні зображення у вигляді Q – спектра та їх спрощення при перед обробці корелюємих зображень по адаптивному методу узагальненого контурного препарування. Адаптивність методу заключається в настройці порогу препарування по параметрам просторового розподілення яскравості вхідного зображення. Скорочення об'єму обчислень заключається в кодуванні препаратів попереднього зображення і заміні операцій множення на операції порівняння і підсумовування.

Суть висловлюваного метода полягає у визначенні фрагмента зображення, що включає в себе найбільш інформативну (у відповідності з критерієм, що варіює) ділянку фона, і кореляційно-екстремальному вимірюванні міжкадрового зсуву даного фрагмента з подальшим прирівнюванням виміряного зсуву шуканому зсуву фону, при цьому розміри фрагмента є проміжними між розміром динамічного об'єкта і протяжного фона.

Формування узагальненого контурного препарату на другому кроці алгоритму непринципове. Можливе застосування і передоброблення по методу рангової знакової дельта-модуляції. Однак, так як в цьому випадку $\overset{\circ}{Q}{}^{r}{}_{f\xi}(\tau)=0$ (значення $\overset{+}{Q}{}^{r}{}_{f\xi}(\tau)=0$ і $\overset{-}{Q}{}^{r}{}_{f\xi}(\tau)=0$ не змінюються), необхідне відповідне коректування формул.

Навчання по ЛРП закінчується при досягненні заданої похибки $\varepsilon>0$.

Розглянемо випадок двомірних зображень, для якого:

$$m_f = m_f x \cdot m_f y;\ m_\xi = m_\xi x \cdot m_\xi y \quad \vec{r}=(r_x, r_y);\ \vec{\tau}=(\tau_x, \tau_y) \tag{1}$$

Даний випадок тривіальним чином зводиться до випадку одномірних зображень, якщо еталонне зображення представити у вигляді сукупності з $(m_{\xi y} - r_y)$ строкових еталонних зображень, кожне з яких має довжину $(m_{\xi x} - r_x)$ але не обов'язково, щоб строкові еталонні зображення були однаковими.

Якщо виконуються умови:

$$0 \le r_x \le m_{fx} - r_x; \quad 0 \le \tau_y \le m_{fy} - r_y \quad m_f x \ge m_\xi x; \ \vec{\tau} = (\tau_x, \tau_y), \tag{2}$$

то легко перейти до процесу навчання

$$P_{H1} = \frac{m_\xi x \cdot m_\xi y}{3} = \frac{m_\xi}{3}.$$

З цього можна зробити висновок, що для випадку двовимірних зображень, з ціллю зменшення діапазону відліків рангової взаємно-кореляційної функції (РВКФ), а також максимального зняття додаткових обмежень доцільно вибирати на етапі навчання

$$\vec{r} = (1,0)$$
$$m_\xi y = 1 \tag{3}$$

при чому можна обмежитись обчисленням неповного поля відліків РВКФ, наприклад, $m_f' y$ рядками так, щоб виконувалось

$$m_f' y \cdot m_f x \ge 3^{m_\xi x} P_{äîâ} K\sigma. \tag{4}$$

Не зважаючи на таку перевагу алгоритму, схема якого представлена на рис.2, як одноманітність процесу навчання по локально-різницевому порогу (ЛРП) з процесом роботи (відмінність в еталонних зображеннях), даний алгоритм вже нескладний в застосуванні і може бути суттєво спрощений.

Спрощення засноване на тому, що для поточного шумового зображення має місце

$$m_{f_1 0} = m_f \cdot P_0$$
$$0{,}5(m_{f_{1+}} + m_{f_{1-}}) = m_f \left(\frac{1 - P_0}{2}\right) = m_f P_n \tag{5}$$

Підставимо в (5) вираз

$$m_f \approx \left(\frac{1}{1} - P_{дов} \mathrm{K}\sigma\right) + \mathrm{r} \approx \frac{1}{1} - P_{дов} K\sigma \cdot m_f^r$$
$$P_0 \cong \begin{cases} 0{,}5\left(1 - \sqrt{\frac{2}{\pi}}\psi_H\right) при \psi_H \in (\psi_{H.опт} - \varepsilon\psi_{H.опт}) \\ \sqrt{\frac{2}{\pi}}\psi_{\mathrm{H}} при \psi_H \in (\psi_{H.опт} : \psi_{\mathrm{H.опт}} + \varepsilon), \end{cases} \tag{6}$$

де $\varepsilon > 0,\ \varepsilon \to 0$

отримаємо:

$$\Delta\psi_H \begin{cases} \sqrt{2\pi}\Delta m^r_{f_1 0}\left(1 - P_{\partial o в} K\sigma\right) npu \Delta \mathrm{m}_{\mathrm{f}_1 0} > 2\varepsilon \\ -\sqrt{\frac{\pi}{2}}\Delta m^r_{f_1 0}\left(1 - P_{\partial o в} K\sigma\right) npu \Delta \mathrm{m}_{\mathrm{f}_1 0} < -\varepsilon \end{cases} \tag{7}$$

де $\Delta m^r_{f_1 0} = m^r_{f_1 0} - m^r_{f/3}, \varepsilon > 0 \ \ (\varepsilon \to 0)$

Розрахункові дані, що пояснюють величину коефіцієнтів в (6) наведені в табл.1.

Таблиця 1

Показник	Значення		
$P_{äîâ} K\sigma$	0.6827	0.9545	0.9973
$\sqrt{2\pi}\left(1 - P_{äîâ} K\sigma\right)$	0.8	0.114	0.0068
$\sqrt{\frac{\pi}{2}}\left(1 - P_{äîâ} K\sigma\right)$	0.4	0.057	0.0034

Висновок. Таким чином, процес навчання ЛРП звівся до знаходження середнього $\tilde{\Delta} f^r$ модуля вибіркових середніх значень позитивних і негативних різниць в різницевому шумовому або фоновому зображенні й установці ЛРП $\Psi = \tilde{\Delta} f^r$. При цьому, враховуючи стаціонарність шуму у сусідніх кадрах зображення, для економії пам'яті, знаходження оцінки $\tilde{\Delta} f^r$ доцільно об'єднати з вводом поточного кадру зображення, а поріг $\Psi = \tilde{\Delta} f^r$ використовувати при попередній обробленні даним методом слідуючи за ним кадрів (число яких визначається часом стаціонарності шуму або фону відповідно).

ЛІТЕРАТУРА

1. Акаев А. А. Оптические методы обработки информации / Акаев А. А., Майоров С. А. – М. : Высш. шк., 1988. – 214 с.

2. Хуан К. Перспективные методы параллельной обработки и архитектура супер-ЭВМ / Хуан К. — ТИИЭР, 1987. – т. 75. – № 10.

3. Комарцова Л. Г. Нейрокмпьютеры / Комарцова Л. Г., Максимов А. В.– М. : МГТУ, 2002. – 320 с.

4. Русин Б. П. Системи синтезу, обробки та розпізнавання складноструктурованих зображень / Русин Б. П. – Львів : Вертикаль, 1997. - 264 с.

5. Особливості побудови оптико-електронних засобів око-процесорного типу для оброблення зображень / Кожем'яко В. П., Поплавський А. Б., Павлов С. В., Бурденюк І. І. // Оптико-електронні інформаційно-енергетичні технології. – 2006. – №1 (11). – С. 47 – 54.

6. Аналіз методів виділення контурів при обробленні біомедичних зображень / Кожем'яко В. П., Павлов С. В., Бурденюк І. І., Поплавський А. Б. // Оптико-електронні інформаційно-енергетичні технології. – 2006. – №2 (12). – С.185 – 191.

7. Оптико-електронні інформаційні технології контролю реологічних властивостей крові / Кожем'яко В. П., Павлов С. В., Бурденюк І. І., Ганиш Н. В. // Інформаційні технології комп'ютерної інженерії. – 2007. – № 1. – С. 75 – 80.

8. Підвищення вірогідності контролю реологічних властивостей крові при застосуванні оптико-електронних кореляційно-екстремальних засобів. / Кожем'яко В. П., Ганиш Н. В., Бурденюк І. І., Поплавський О. А. // Оптико-електронні інформаційно-енергетичні технології. – 2007. – № 1(13). – С.139–145.

9. Оптико-електронні методи контролю біомедичних зображень / Кожем'яко В. П., Ганиш Н. В., Бурденюк І. І., Білоконний В. Д. // Оптико-електронні інформаційно-енергетичні технології. – 2007. – №2 (14). – С. 198 – 208.

10. S. V. Pavlov; V. B. Vassilenko; I. R. Saldan; D. V. Vovkotrub; A. A. Poplavskaya, et al. Methods of processing biomedical image of retinal macular region of the eye, *Proc. SPIE* 9961, Reflection, Scattering, and Diffraction from Surfaces V, 99610X (September 26, 2016); doi:10.1117/12.2237154

11. Oleg G. Avrunin; Maksym Y. Tymkovych; Sergii V. Pavlov; Sergii V. Timchik; Piotr Kisała, et al. Classification of CT-brain slices based on local histograms, *Proc. SPIE* 9816, Optical Fibers and Their Applications 2015, 98161J (December 18, 2015); doi:10.1117/12.2229040

12. Olexander N. Romanyuk; Sergii V. Pavlov; Olexander V. Melnyk; Sergii O. Romanyuk; Andrzej Smolarz, et al. Method of anti-aliasing with the use of the new pixel model , *Proc. SPIE* 9816, Optical Fibers and Their Applications 2015, 981617 (December 18, 2015); doi:10.1117/12.2229013

13. Vladimir V. Kholin; Oksana M. Chepurna; Irina O. Shton; Valerii S. Voytsehovich; Olexiy D. Azarov, et al. Methods and fiber optics spectrometry system for control of photosensitizer in tissue during photodynamic therapy, *Proc. SPIE* 10031, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2016, 1003138 (September 28, 2016)

14. O N. Romanyuk; S. V. Pavlov; R. Yu. Dovhaliuk; N. P. Babyuk; M. D. Obidnyk, et al.Microfacet distribution function for physically based bidirectional reflectance distribution functions ", *Proc. SPIE* 8698, Optical Fibers and Their Applications 2012, 86980L (January 11, 2013); doi:10.1117/12.2019338

15. Vassilenko, S Valtchev, JP Teixeira, S Pavlov. Energy harvesting: an interesting topic for education programs in engineering specialities / «Internet, Education, Science" (IES-2016) – 2016. – P. 149-156.

Отримано 26.06.2017

УДК 004. 925.9

ВИКОРИСТАННЯ МЕТОДУ УЗАГАЛЬНЕНОГО Q-ПЕРЕТВОРЕННЯ ЗОБРАЖЕННЯ В ЗАДАЧАХ ОБРОБЛЕННЯ НОРМАЛЬНОГО БІЛОГО ШУМУ

Павлов С.В., Тимченко Л.І., Бурденюк І.І.,

Чернишова Т.А., Поплавська А.А.

Вінницький національний технічний університет

The article shows that the extremum of dependence corresponds to the maximum of the so-called total absolute stock of the reference image, which minimizes the average amplitude of the erroneous peak of the total VCR of the OCRs of the reference and current image when working on arbitrary backgrounds. The influence of the confidence probability of the learning process on the choice of the main parameters of pre-processing: the lengths of the producing images, the magnitudes of the local multi-threshold threshold.

Актуальність. Вибір величини ψ локального різницевого порогу (ЛРП) з ціллю максимального придушення адитивного нормального білого шуму в зображенні є актуальним при узагальненому Q-препаруванні (УQП) зображення або після його часткового Q-сумування (в останньому випадку при створенні широкопольних кореляторів). Частково, для випадку кореляційного виявлення сигналу постійної амплітуди з наявністю вказаного виду шума.

Метод. В роботі даному викладені теоретичні основи процесу вивчення корелятора з передобробкою зображень за методом ОQП, що проводиться з ціллю вибору величини $\psi_{ОПТ}$. ЛРП, що мінімізує середню величину відліків рангової взаємної кореляційної функції (РВКФ) $Q_{fg}(\vec{\tau})$ еталонного $G(i', j') = \{g_{i'}, j'\}$ і поточного $F(i, j) = \{f_i, j\}$ зображення, що представляє собою реалізацію нормального білого шума.

Графік щільності ймовірності P(U) розподілення амплітуди білого шума наведений на рис.1, а,б

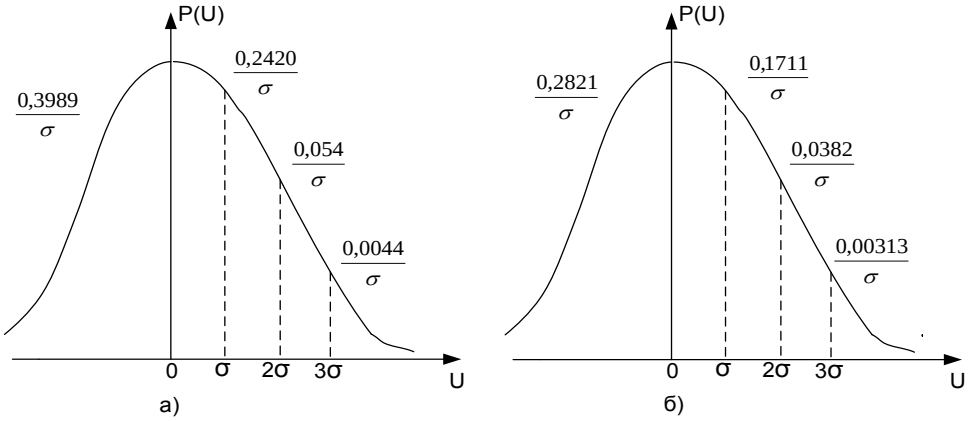


Рисунок 1 - Графік щільності ймовірності P(U)
а) графік щільності імовірності P(U) розподілення амплітуди білого шума;
б) графік щільності імовірності розподілення різниці відліків білого шума

$$P(U) = \frac{1}{\sqrt[\sigma]{2\pi}} \cdot e - \frac{U^2}{2B^2} \qquad (1)$$

де σ- середньоквадратичне відхилення (СКВ) амплітуди шумового сигналу.

Відмітимо, що формула (1) справедлива для описання шума телевізійного фотоприймального пристрою, але не може використовуватися для описання одно-полярного сигналу від випадкового фонового зображення (фону) із нормальним законом розподілення щільності ймовірності яскравості.

Закон розподілення щільності ймовірності P(ΔU) різності ΔU відліків нормального білого шума, представлений на рис.1,б має наступний вигляд:

$$P(\Delta U) = \frac{1}{\sqrt[\sigma]{2\pi \cdot \sqrt{2\pi}}} \cdot e - \frac{\Delta U^2}{2(\sqrt[\sigma]{2})^2} \qquad (2)$$

Даний закон справедливий і для різниці відліків сигналу від випадкового фону з нормальним законом розподілення щільності ймовірності яскравості. Тому висновки, зроблені в подальшому відносно придушення вкладу адитивного нормального білого шуму в РВКФ можуть бути розширені і на придушення вкладу сигналу випадкового фону в цю кореляційну функцію.

1. Показано, що:

- залежність $P_{H1}(P_0)$ середньої величини підрахунків рангової ВКФ (РВКФ) ОКП-тів еталонного і поточного зображення, що являють собою реалізації нормального білого шуму, від

ймовірності P_0 формування відліків нульового ОКП-та, що в свою чергу залежить від ЛРП Ψ , має чітко виражений екстремум (мінімум) в точці $P_0 = P_{00} = 0{,}33$; $P_{H1}(P_0)$ – екстремум $P_{H1}(P_0)$ відповідає рівності $\overset{+}{m}_\xi = \overset{0}{m}_\xi = \overline{m}_\xi = m_\xi / 3$ чисел одиничних відліків позитивного, нульового і негативного ОКП-тов як для еталонного, так і для поточного зображення, причому величина екстремуму $P_{H1}(P_0)$ рівна одній третині числа загального відліків ОКП-та еталонного зображення;

- залежність $P_{H1}(P_0)$ в області екстремуму $P_0 = P_{00}$ близька до асимптот $P_{H1}(P_0) = m_\xi P_0$ і $P_{H1}(P_0) = 0.5 m_\xi (1 - P_0)$;
- в області екстремуму $P_0 = P_{00}$ залежності $P_{H1}(\Psi)$ близькі до асимптот $P_{H1}(\Psi) = 0.5 m_\xi (1 - \Psi\sqrt{2/\pi})$ і $P_{H1}(\Psi) = m_\xi \sqrt{2/\pi}$;
- для екстремуму залежностей $P_{H1}(P_0)$ або $P_{H1}(\Psi)$ величина $\Psi_{H} opm$ нормованого відносно СКО σ_Δ ЛРП складає 0,33 $\sqrt{\pi/2}$, де σ_Δ - СКО різницевого шумового зображення;
- екстремум залежності $P_{H1}(P_0)$ відповідає максимуму так званого сумарного абсолютного запасу еталонного зображення, що забезпечує мінімізацію в середньому амплітуди хибного піку сумарної ВКФ ОКП-тів еталонного і поточного зображення при роботі на довільних фонах.

2. Розроблення алгоритму процесу навчання, мінімізуючи середню величину відліків РВКФ еталонного і поточного шумових зображень, попередньо оброблених за методом ОКП. Один із даних алгоритмів базується на знаходженні РВКФ і потребує малих змін роботи ОКП-корелятора.

3. Розробленні методи оцінки СКО σ_Δ, необхідні для отримання абсолютної величини оптимального ЛРП $\Psi opm = \Psi_{H1} opm \sigma_\Delta$ і його регулювання в процесі навчання.

4. Показано існування оцінки $\tilde{\Delta} f^r \approx \Psi opm$, потребуючої мінімальних обчислень і рівному середньому модулів вибіркових середніх значень позитивних і негативних різниць поточного зображення.

5. Досліджено вплив довірчої ймовірності процесу навчання на вибір основних параметрів попередньої обробки: довжин препаруючих зображень, величини ЛРП.

Результати досліджень. На рис. 2 представлено результати оброблення біомедичних зображень на прикладі контролю реологічних властивостей крові.

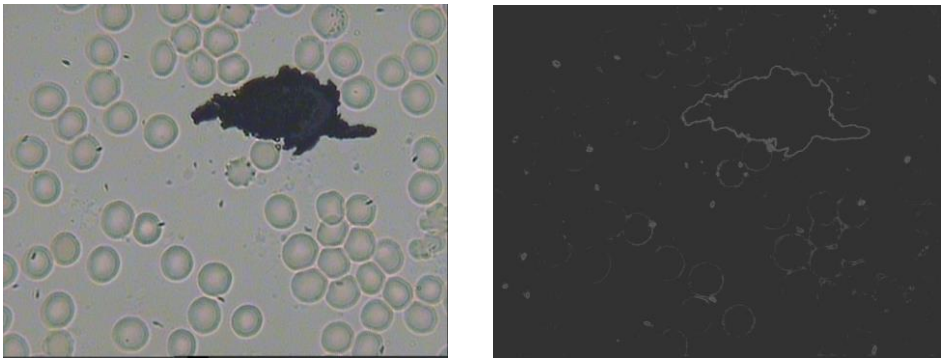


Рисунок 2 - Результат оброблення зображення отриманого за допомогою ОЕКЕЗ
а) вихідне зображення в ОЕКЕЗ; б)Зображення отримане за допомогою ОЕКЕЗ (алгоритм SUSAN)

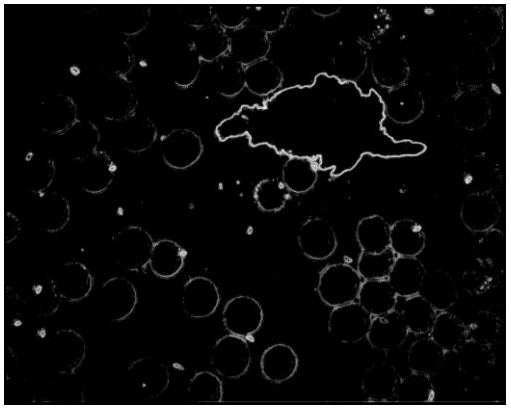


Рисунок 2,б. Результат використання поелементарної обробки зображення (покращення субэктивного сприйняття)

Показано, що залежність $P_{H1}(P_0)$ середньої величини підрахунків рангової ВКФ (РВКФ) ОКП-тів еталонного і поточного зображення, що являють собою реалізації нормального білого шуму, від ймовірності P_0 формування відліків нульового ОКП-та, що в свою чергу залежить від ЛРП Ψ , має чітко виражений екстремум (мінімум) в точці $P_0 = P_{00} = 0{,}33$.

Висновок. Показано, що екстремум залежності $P_{H1}(P_0)$ відповідає максимуму так званого сумарного абсолютного запасу еталонного зображення, що забезпечує мінімізацію в середньому амплітуди хибного піку сумарної ВКФ ОКП-тів еталонного і поточного зображення при роботі на довільних фонах. Досліджено вплив довірчої ймовірності процесу навчання на вибір основних параметрів попередньої обробки: довжин препаруючих зображень, величини ЛРП.

ЛІТЕРАТУРА.

1. Акаев А. А. Оптические методы обработки информации / Акаев А. А., Майоров С. А. – М. : Высш. шк., 1988. – 214 с.
2. Хуан К. Перспективные методы параллельной обработки и архитектура супер-ЭВМ / Хуан К. — ТИИЭР, 1987. – т. 75. – № 10.
3. Комарцова Л. Г. Нейрокмпьютеры / Комарцова Л. Г., Максимов А. В.– М. : МГТУ, 2002. – 320 с.
4. Русин Б. П. Системи синтезу, обробки та розпізнавання складноструктурованих зображень / Русин Б. П. – Львів : Вертикаль, 1997. - 264 с.
5. Кожем'яко В.П., Павлов С.В., Ганиш Н.В., Бурденюк І.І. Оптико-електронні інформаційні технології контролю реологічних властивостей крові // Інформаційні технології та комп'ютерна інженерія. – 2007. - №2(9). – С. 75-80.
6. Кожем'яко В.П., Ганиш Н.В., Бурденюк І.І., Поплавський А.В. Підвищення вірогідності контролю реологічних властивостей крові при застосуванні оптико-електронних кореляційно-екстремальних засобів // Оптико-електронні інформаційно-енергетичні технології. – 2007. - №1(13). – С. 139-145.
7. S. V. Pavlov; V. B. Vassilenko; I. R. Saldan; D. V. Vovkotrub; A. A. Poplavskaya, et al. Methods of processing biomedical image of retinal macular region of the eye, *Proc. SPIE* 9961, Reflection, Scattering, and Diffraction from Surfaces V, 99610X (September 26, 2016); doi:10.1117/12.2237154
8. Oleg G. Avrunin; Maksym Y. Tymkovych; Sergii V. Pavlov; Sergii V. Timchik; Piotr Kisała, et al. Classification of CT-brain slices based on local histograms, *Proc. SPIE* 9816, Optical Fibers and Their Applications 2015, 98161J (December 18, 2015); doi:10.1117/12.2229040
9. Olexander N. Romanyuk; Sergii V. Pavlov; Olexander V. Melnyk; Sergii O. Romanyuk; Andrzej Smolarz, et al. Method of anti-aliasing with the use of the new pixel model , *Proc. SPIE* 9816, Optical Fibers and Their Applications 2015, 981617 (December 18, 2015); doi:10.1117/12.2229013
10. Vladimir V. Kholin; Oksana M. Chepurna; Irina O. Shton; Valerii S. Voytsehovich; Olexiy D. Azarov, et al. Methods and fiber optics spectrometry system for control of photosensitizer in tissue during photodynamic therapy, *Proc. SPIE* 10031, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2016, 1003138 (September 28, 2016)
11. O N. Romanyuk; S. V. Pavlov; R. Yu. Dovhaliuk; N. P. Babyuk; M. D. Obidnyk, et al. Microfacet distribution function for physically based bidirectional reflectance distribution functions ", *Proc. SPIE* 8698, Optical Fibers and Their Applications 2012, 86980L (January 11, 2013); doi:10.1117/12.2019338
12. Vassilenko, S Valtchev, JP Teixeira, S Pavlov. Energy harvesting: an interesting topic for education programs in engineering specialities / «Internet, Education, Science" (IES-2016) – 2016. – P. 149-156.

Отримано 26.06.2017

УДК 004. 925.9

ТРИВИМІРНЕ МОДЕЛЮВАННЯ ДЛЯ ПЛАНУВАННЯ ТА ПРОВЕДЕННЯ ПЛАСТИЧНИХ ОПЕРАЦІЙ НА ОБЛИЧЧІ ЛЮДИНИ

Романюк С.О,
Вінницький національний технічний університет, Україна,
Безмертний Ю. О., д.мед.н.
Науково-дослідний інститут реабілітації інвалідів , Україна
Безсмертна Г. В., к.м.н,
Науково-дослідний інститут реабілітації інвалідів , Україна

Розглянуто особливості використання тривимірного моделювання для планування та проведення пластичних операцій на обличчі людини. Запропоновано алгоритм формування тривимірної моделі обличчя та комп'ютерної системи для її реалізації.

Пластичні операції в сучасній медицині є невід'ємною частиною відновної та естетичної хірургії. Більшість оперативних пластичних втручань спрямована на зміну зовнішнього вигляду різних частин тіла з метою задоволення як особистих уподобань, так з метою відновлення зовнішності від травм, швів від перенесених операцій і т.д.

За інформацією, наданою Міжнародним товариством пластичної та естетичної хірургії (ISAPS) [1], щорічно в світі проводиться більше 17 млн. пластичних операцій. Найбільше цих операції проводиться в США, Бразилії, Китаї, Мексиці, Індії та Японії. Що стосується операцій на обличчі та голові, то Бразилію посідає перше місце. США вийшли на друге місце .

У 2016 році було досягнуто 9-процентне збільшення порівняно з минулим роком.

У 2016 році в США було виконано 4,2 млн. пластичних операцій [1].

Щодо України, то за останні п'ять років кількість пацієнтів клінік естетичної медицини зросла приблизно на чверть і становить майже 15 тисяч осіб на рік. Незважаючи на те, що якість пластичної медицини в Україні ще не досягло світового рівня, все більше українців шукають пропозицій за кордоном і все частіше цей вибір доводиться на Туреччину.

Пластичні операції на обличчі мають на меті усунення дефектів і деформацій різних органів і ділянок обличчя і відновлення їх форми і функції [2] .

Необхідність в пластиці особи пов'язана з різними дефектами зовнішності - як вродженими, так і набутими в результаті вікових змін, травм або хвороб. Найбільш поширені показання: вікові зміни особи, виражена асиметрія особи, птоз (опущення) верхніх і нижніх повік і м'яких тканин обличчя, провисання шкіри, глибокі зморшки і складки, сліди (рубці і шрами) у результаті перенесених шкірних захворювань, подвійне підборіддя. Як правило, це питання естетики, і за великим рахунком не існує об'єктивних обов'язкових показань до проведення пластичних операцій [2].

У результаті проведення пластичних операцій [4] вдається досягти відразу двох позитивних моментів. Перший полягає у відтворенні необхідної естетичної складової, а другий - робить прямий вплив на психоемоційний стан пацієнта. Успішно проведена процедура допомагає людині позбутися набору певних комплексів, пов'язаних з незадовільним станом будь-яких частин обличчя. У глобальному плані естетична хірургія благотворно впливає на психічну складову пацієнта.

При пластичних операціях [4] на обличчі поряд з завданнями відновлення анатомічної будови і функціональної повноцінністю органу великої уваги приділяється також естетичному фактору - відтворення зовнішніх обрисів обличчя (губи, ніс), пропорційному співвідношенню їх розмірів, відповідності кольору шкіри.

Комп'ютерне 3D моделювання дозволяє візуалізувати уявлення про майбутню операцію для пацієнта, щоб той міг у повній мірі усвідомити, чи справді він потребує цього втручання. Більш того, створена модель дозволить вибрати, яка же дійсно повинна бути ідеальна зовнішність, на його думку.

Комп'ютерне 3D моделювання зовнішності дозволяє побачити себе на моніторі комп'ютера з тими змінами, які відбудуться після операції. Таке моделювання має важливу та незаперечну перевагу перед проведенням операції без попереднього аналізу - це мінімізація ризиків, адже пацієнт реально оцінює результат усіх заходів.

Передбачається, що з часом будь-операційне втручання засобів пластичної хірургії буде попередньо оцінюватися за допомогою побудови 3D моделі [3], адже це виключить можливі розбіжності і невдоволення з боку пацієнтів. Ризик від подібних операцій знизиться до мінімальних значень.

За допомогою новітніх розробок була впроваджена методика 3D моделювання за результатами магнітно-резонансної томографії (МРТ). На основі даних томографії формується модель з візуалізацією твердих і м'яких тканин. Це дозволяє не просто оцінити зовнішній вигляд очікуваного результату, але і врахувати персональні дані, що

дуже важливо для точних попередніх розрахунків. При такому моделюванні можна приміряти імплантати на своїх місцях і гранично точно зрозуміти, як зміниться зовнішність після реального втручання. Очевидно, що це дає більш реальну картину очікуваних змін, на відміну від банального моделювання по фотографії.

Переваг у цій технології дуже багато. Вона дозволяє не обмежувати себе приблизними розрахунками і базуватися на достовірних і повних даних про тканини.

3D моделювання на основі томографії дозволяє оптимізувати і поліпшити процес підготовки до операції, а, отже, і самого втручання, і заходів післяопераційного періоду. І хірург, і пацієнт бачать тривимірне деталізоване зображення, яке точно відображає реальну картину, тому і планування, і проведення необхідних заходів виявляються гранично ефективними і точними.

Система дозволяє швидко та максимально точно визначити площу, обсяг і будь-які необхідні характеристики тканин, що спрощує процес розрахунків. Лікар не витрачає час на зайві виміри, він може присвятити більше часу роботі безпосередньо з клієнтом, що робить весь процес взаємодії якісніше і простіше.

Крім того, 3D-моделювання за допомогою томографії створює всі можливі умови для більш точного підбору імплантатів і протезів, які в більшій мірі задовольнять пацієнта. Якісні та детально опрацьовані імплантати грають одну з найважливіших ролей в процесі зміни тіла, від них залежить те, наскільки природно він себе буде відчувати з ними, а головне, наскільки близько вони будуть відповідати тому, як він їх собі уявляв.

Важливий є той факт, що в разі проведення комп'ютерного 3D моделювання перед операцією контакт між пацієнтом і лікарем буде максимально довірчим і продуктивним.

Тривимірна модель обличчя має такі переваги [5]:

- створюється об'ємна модель, ідентична оригіналу, з урахуванням найменших змін рельєфу складових тканин при мінімальних похибках;
- можливості проведення аналізу та виконання високоточних геометричних вимірювань отриманої моделі будь-якої складності залежно від поставлених задач;
- велика об'єктивність даних порівняно з традиційною фіксацією методом фотографування - якість даних не залежить від освітлення на момент зйомки, вибраних ракурсів або застосовуваної оптики.

Важливо створити гнучку 3D-модель обличчя, що дасть можливість її простото редагувати. Другим перевагою таких моделей є їх компактне представлення: Для цього в клієнт-серверній

архітектурі достатньо задати гнучку модель один раз, а надалі передавати лише набір її параметрів.

Інтенсивне впровадження принтерів тривимірного друку призвело до того, що 3D-друк [6] проникла в усі галізу, починаючи від виробництва простих пластичних деталей і закінчуючи застосуванням у медицині.

Сьогодні найчастіше тривимірний друк використовується для виготовлення імплантатів. Це спеціальні матеріали природного та синтетичного походження, що застосовуються у пластичній хірургії з метою створення необхідних форм і об'ємів з метою поліпшення обрисів різних ділянок людського обличчя, або заміни ушкоджених по різним причинам його частин.

Використання тривимірного моделювання для розробки імплантатів знижує тимчасові витрати, зменшує інтраопераційну травматичність і експозицію рани, а також сприяє більш точному розташуванню імплантатів з урахуванням індивідуальних особливостей кожного клінічного випадку.

У більшості випадків для проведення імплантації в застосовуються біологічно сумісні матеріали синтетичного походження.

Сьогодні найкращі імплантати виробляються з використанням тривимірної графіки, оскільки в цьому випадку можна розробити цільний імплантат, який ідеально підходить конкретному пацієнту. При цьому досягається сумісність, що важко добитися в разі використання звичайних моделей. Використання 3D-технологій у медицині дозволяє скоротити вірогідність помилок до мінімуму та виключити подальші помилки при вживанні імплантату.

Імплантати можуть бути виготовлені з цілого ряду матеріалів, включаючи живі органічні клітини. Вибір того чи іншого матеріалу для медичного прототипування залежить від цілей і завдань, що стоять перед медиками, та проблем, пов'язаних із здоров'ям пацієнта.

Можливо виготовлення макету хворого органу для проведення більш детальної діагностики та підготовки до операції.

Людські органи друкують і з метою створення експериментальної моделі. У випадку операції відтворюють 3D-модель [3] органу, що вимагає хірургічного втручання. Хірург тренується спочатку на моделі, а потім проводить операцію.

Передопераційне планування з використанням тривимірної комп'ютерної реконструкції і моделювання підвищує точність проведення оперативних втручань у пацієнтів, дає можливість для більш детального вивчення конкретного клінічного випадку, сприяючи прийняттю найбільш прийнятного рішення про тактику

подальшого лікування. Це знижує тимчасові витрати, зменшує інтраопераційну травматичність і експозицію рани, а також сприяє більш точному розташуванню імплантатів з урахуванням індивідуальних особливостей

По данным Oxford Performance Materials [7], в найближчі роки більше 450 тис. пацієнтів по всьому світу отримають друковані органи (в цю програму інвестовано близько 2 млрд доларів). Вони краще підходять, ніж донорські, так як виготовлені з кліток рідного органу людини і повністю виключають ризик відторження.

У даний момент успішно пересаджують вуха (роздруковується раковина, в яку вкладають слуховий апарат), трахеї (виготовляють з пластикових волокон і живих кліток), міжпозвонкові диски, фрагменти грудної клітки, частини печінки та серця,. В США в опікових центр ах масово використовують для пересадки друковану шкіру.

Технічний директор компанії Google Рэй Курцвейл [8] прогнозує, що масовий та доступний 3D-друк стане після 2030 року. У своєму прогнозі він пише, що через п'ять років 3D-принтери будуть централізовано постачатися в лікарні, а заміна пошкоджених органів на нові дозволить істотно продовжити життя людей. Так само, по прогнозам Курцвейл, через 15 років з'являється технологія, що дозволить створювати комплекси для друку органів безпосередньо всередині пацієнта [10].

Якісне навчання будь-якого спеціаліста, що працює в галузі медицини, неможливо без великої кількості практичних занять. Для цих цілей використовувалися мертві людські тіла. Сьогодні пропонується використовувати в якості об'єктів для практичних занять моделі людських м'язів, кісток і внутрішніх органів, створених за допомогою 3D-принтера. Тепер майбутні хірурги проводять свої перші справжні операції на елементах людських тіл, отриманих за технологією 3D-прототипування.

3D-графіка широко використовується в медицині для діагностування [9]. Англійські вчені створили технологію, яка дозволяє виявляти генетичні відхилення у дітей у ранньому віці, шляхом аналізу сканованого зображення особи у форматі 3D. Дана технологія дозволяє виявляти до 700 відхилень. Спочатку проводять об'ємне сканування лиця особи. Після цього отримані результати порівнюються з інформаційною базою, в якій зберігаються 3D моделі. Точність діагностування даної технології становить 90%.

Другою корисною можливістю застосування 3D друку в медицині, стала діагностика. Завдяки 3D сканування лікар отримує найточнішу модель органу і, завдяки 3D-друку, може створити повну

копію. По прогнозам учених, вже до 2020 року людство буде здатне друкувати будь-який орган. Така технологія дозволить відмовитись від донорства. Це може допомогти в лікуванні різних хвороб, наприклад, раку, просто замінюючи пошкоджені органи та тканини новими, друкованими.

Таким чином, тривимірне моделювання для планування та проведення пластичних операцій забезпечує принципово новий рівень надання послуг.

ЛІТЕРАТУРА

1. ISAPS Global Statistics/ [Електронний ресурс] Режим доступу: http://www.isaps.org/news/isaps-global-statistics .
2. Виды (классификация) пластических операций на теле [Електронний ресурс] / Режим доступу: http://ru-ideal.com/plast.
3. Романюк О. Н. Високопродуктивні методи та засоби зафарбовування тривимірних графічних об'єктів. Монографія. / О. Н. Романюк, А. В. Чорний. - Вінниця : УНІВЕСУМ-Вінниця, 2006. — 190 с.
4. Пластика лица: виды, особенности процедуры и ее стоимость. [Електронний ресурс] / Режим доступу: https://www.kp.ru/guide/plastika-litsa.html/.
5. Лесняков А.Ф. Возможности применения технологии 3D-сканирования для объемной визуализации в пластической хирургии лица / А.Ф Лесняков, М.А Волох., М.А Кирилловский// Вестник Северо-западного государственного медицинского университета им. И.И. Мечникова. Том: 8. № 1.С. — 2016. — 7-12.
6. Романюк С. О. Використання 3D принтерів у медичній практиці / С. О. Романюк, В. М.Чорний // MATERIALY XII MIEDZYNARODOWEJ NAUKOWI-PRAKTYCZNEJ KONFERENCJI «NAUKOWA PRZESTRZEN EUROPY - 2016» 07-15 kwietnia. — 2016. — Vol. 12. Przemysl. Nauka I studia. — P. 28—33.
7. Органи друку. медики почали створювати органи для трансплантації на 3d-принтерах [Електронний ресурс]. Режим доступу: http://newsonline24.com.ua/organi-druku-mediki-pochali-stvoryuvati-organi-dlya-transplantaci%D1%97-na-3d-printerax/
8. Технический директор Google: прогноз до 2099 года [Електронний ресурс]. Режим доступу: http://mmr.ua/show/tehnicheskij-direktor-google-raspisal-buduschee-mira-prognoz-do-2099-goda/44073#1642997652.1504442307.
9. Диагноз путем анализа сканированного изображения лица в формате 3D [Електронний ресурс]. Режим доступу: http://netrmed.livejournal.com/tag/%D0%B4%D0%B8%D0%B0%D0%B3%D0%BD%D0%BE%D0%B7
10. В Украине начали печатать части тела на 3D-принтере [Електронний ресурс]. Режим доступу: http://vesti-ukr.com/nauka-i-tehnologii/117560-v-ukraine-nachali-pechatat-chasti-tela-na-3d-printere

Отримано 26.06.2017

УДК 004. 925.9

АНАЛІЗ ГРАФІЧНОЇ АХІТЕКТУРИ NVIDIA PASCAL

Романюк О. Н, Романюк О. В., Войт Б. Л.
Вінницький національний технічний університет, Україна

Проаналізовано особливісті архітектури NVIDIA PASCAL. Розглянуто організацію інтерфейсу, багатошарової пам'яті, технологію багатобуферного формування зображень, технологію мультипроектування зображень.

Графічні процесори забезпечують формування реалістичних графічних сцен з високою продуктивністю [1].

Графічні процесори NVIDIA нового покоління відомі під кодовим ім'ям Pascal [2-4] і оприлюднені у 2016 році. Відеокарти на базі архітектури Pascal забезпечують високу продуктивність і енергоефективність. Вони створені з використанням надшвидкісної технології FinFET [5] і підтримують DirectX ™ 12 [6] для швидкого, плавного та енергоефективного обчислювального процесу.

Володіючи 16 млрд. транзисторів, побудованих на базі 16-нм техпроцесу FinFET, графічний процесор Pascal є найбільшим в світі FinFET чіпом. Він створений для забезпечення високої продуктивністі та енергоефективності для задач з практично необмеженими обчислювальними вимогами.

Однією з особливостей Pascal є потрійний буфер для виведення кадрів, що забезпечує одночасно наднизькі затримки в відображенні та забезпеченні вертикальної синхронізації. В одному буфері зберігається виведене зображення, в іншому - останній відформований кадр, у третьому - малюється поточний.

Чіпи на базі архітектури Pascal підтримують багатошарову пам'ять з високою пропускною здатністю (stacked DRAM за класифікацією NVIDIA, конкретний тип - SK Hynix high-bandwidth memory (HBM) [7] або Micron hybrid memory cube, уніфікований адресний простір пам'яті для CPU і GPU, шину NVLink для зв'язку з CPU і GPU у системах для суперкомп'ютерних обчислень, а також цілий ряд нових графічних, обчислювальних і мультимедійних функцій, які будуть підтримуватися DirectX 12, OpenGL 5 [8] і іншими майбутніми програмними інтерфейсами.

NVIDIA реалізувала об'єднання процесора і GPU, Вони зможуть користуватися ресурсами один одного. Це дозволить краще розподіляти навантаження в DX12- середовищі.

Pascal - це перша архітектура, оснащена високошвидкісним двонаправлен

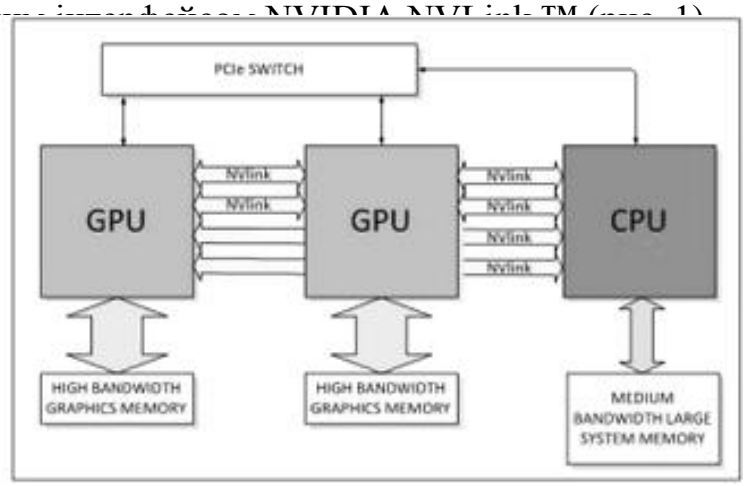


Рисунок 1 - Інтерфейс NVIDIA

Ця технологія дозволяє масштабувати додатки на кількох GPU, забезпечуючи п'ятикратний приріст продуктивності порівняно з кращими на сьогодні рішеннями в своєму класі.

Технологія 3D Memory або Stacked Memory усуває недоліки зв'язку між GPU і відеопам'яттю. Інтерфейс NVLink покликаний збільшити ефективність з'єднання між GPU і CPU, а також між декількома GPU.

У даному відношенні на першому місці знаходиться пропускна здатність, яка сьогодні забезпечується поширеним інтерфейсом PCI Express. 16 ліній PCI Express 3.0 забезпечують пропускну здатність 15,75 Гбайт/с або 128 GT/s. NVIDIA, наприклад, з архітектурою "Maxwell" додано технологію стиснення пам'яті, яка покликана задовольнити зростаючі вимоги до пропускної здатності пам'яті.

NVIDIA використовує для NVLink з'єднання «точка-точка». Одне з'єднання NVLink охоплює вісім ліній. У Pascal буде використовуватися чотири з'єднання NVLink. За інформацією NVIDIA, кількість з'єднань може змінюватися залежно від цільового ринку. З'єднання NVLink можна гнучко комбінувати, якщо того вимагають сценарії використання.

Основа для NVLink - швидкодіючий, 8-смуговий, подвійний двунаправлений зв'язок. Pascal GPUs підтримує багато цих зв'язків, забезпечуючи гнучкість конфігурації. Відеокарти можуть сполучатися разом (або відеокарта і процесор), щоб сформувати єдиний зв'язок

GPU↔CPU або використовуватися індивідуально, щоб створити мережу GPU↔CPU або зв'язок GPU↔GPU, що дозволяє використовувати обчислювальні елементи разом, для спільного вирішення різного роду завдань. Якщо центральний процесор не підтримує функцію NVLink, то можна просто об'єднувати GPU.

NVLink більш ніж в два рази ефективніше PCIE 3.0 [9].

Архітектура Pascal задовольняє вимоги дисплеїв нового покоління, включаючи VR-дисплеї і дисплеї ультрависокої роздільної

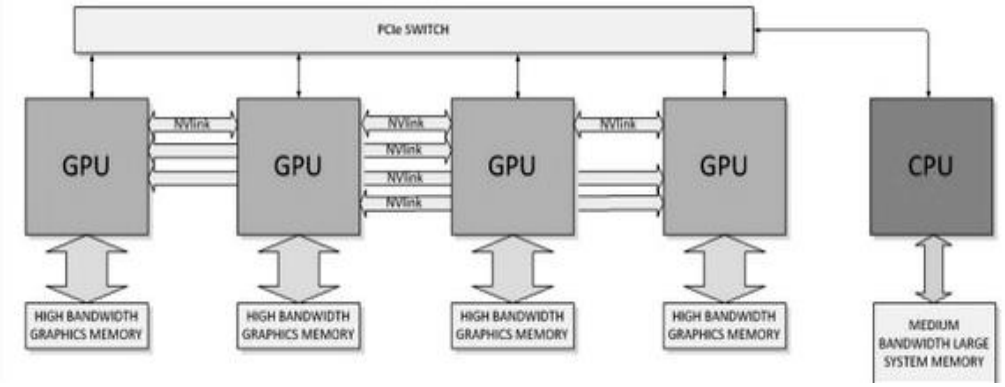


Рисунок 2 – Шина NVLink

здатності, підтримуючи підключення декількох моніторів. Вона оснащена технологіями NVIDIA GameWorks ™, які забезпечують плавний і кінематографічний ігровий процес. Крім того, вона підтримує революційну технологію для запису зображення з охопленням в 360 градусів.

У Pascal модифікували ту частину, яка відповідає за геометричну складову картинки. Це необхідно для мультимоніторних конфігурацій і роботи з VR-шоломами: при належній підтримці з боку ігрового двигуна – відеокарта може один раз порахувати геометрію і отримати кілька проекцій геометрії для кожного з екранів. Це істотно знижує навантаження в VR не тільки при роботі з трикутниками (тут досягається дворазовий приріст), але і в роботі з піксельною складовою.

У Pascal з'явилася апаратна підтримка технології мультипроектування, яка дозволяє працювати одночасно з різними проекціями зображення (рис. 3).

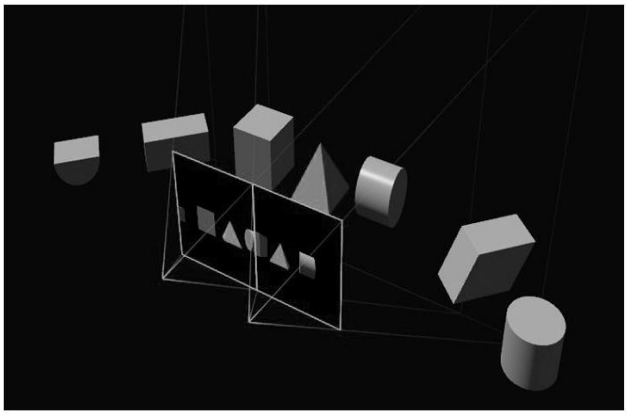


Рисунок 3. – Приклад проекції зображень з використанням технології мультипроектування

Спеціальний блок Simultaneous Multi-Projection всередині PolyMorph Engine відповідає за формування різних проекцій при обробці одного потоку геометрії. Цей блок обробляє геометрію одночасно для 16 проекцій з одним або двома центрами перспективи. Це не вимагає повторної обробки геометрії і дозволяє реплікувати дані до 32 разів (16 проекцій на дві точки).

Рішення, закладені в Kepler і продовжені в поколінні Maxwell, залишилися і в Pascal: перші споживчі відеокарти випущені на базі чіпа GP104, який складається з чотирьох кластерів обробки графіки, повнорозмірний, шестикластерний GP100 під маркою TITAN X [10].

З моменту появи GDDR5 [11] компанія NVIDIA працювала над наступним поколінням швидкісної пам'яті. Результатом взаємодії з розробниками пам'яті стала поява GDDR5X зі швидкістю передачі даних 10 Гбіт/с. Робота з настільки швидкою пам'яттю висуває нові вимоги до розведення електричних ланцюгів. Тому були перероблені лінії передачі даних між GPU і мікросхемами пам'яті, змінено структуру самого чіпа. Все це дозволяє ефективно працювати з надшвидкісним відеобуфером. Серед переваг GDDR5X-більш низька робоча напруга на рівні 1,35 В.

GeForce підтримують спеціальні алгоритми стиснення даних у пам'яті, що дозволяє більш ефективно використовувати кеш і передавати більше даних при тій же пропускній здатності. Підтримується кілька методів, залежно від типу даних вибирається свій алгоритм стиснення. Важливу роль відіграє алгоритм стиснення кольору delta color compression. Завдяки йому кодується не колір

кожного окремого пікселя, а різниця між пікселями при послідовній передачі даних. Обчислюється усереднений колір тайла і дані про усунення кольору для кожного пікселя цього тайла.

ЛІТЕРАТУРА

1. Романюк О.Н. Класифікація графічних відеоадаптерів / О. Н. Романюк, Р. Ю. Довгалюк, С. В. Олійник // Наукові праці Донецького національного технічного університету .Серія Інформатика, кібернетика та обчислювальна техніка". Випуск 14(188). 2011. С. — 211-215.
2. Nvidia Pascal Architecture Overview [Електронний ресурс] Режим доступу: https://www.notebookcheck.net/Nvidia-Pascal-Architecture-Overview.165493.0.html.
3. Архитектура NVIDIA PASCAL [Електронний ресурс] Режим доступу: http://www.nvidia.com.ua/object/gpu-architecture-ru.html.
4. Nvidia Pascal: обзор архитектуры [Електронний ресурс] Режим доступу: https://occlub.ru/testodrom-mobile/2342-nvidia-pascal-obzor-arkhitektury?showall=1.
5. FinFET [Електронний ресурс] Режим доступу: https://en.wikipedia.org/wiki/Multigate_device.
6. DirectX 12 [Електронний ресурс] Режим доступу: https://ru.wikipedia.org/wiki/Direct3D_12.
7. High Bandwidth Memory [Електронний ресурс] Режим доступу: https://ru.wikipedia.org/wiki/High_Bandwidth_Memory.
8. Open Graphics Library[Електронний ресурс] Режим доступу: https://en.wikipedia.org/wiki/OpenGL.
9. PCI Express[Електронний ресурс] Режим доступу: https://ru.wikipedia.org/wiki/PCI_Express.
10. NVIDIA TITAN X: большой Pascal [Електронний ресурс] Режим доступу: https://3dnews.ru/939245.
11. Graphics Double Data Rate/https: [Електронний ресурс] Режим доступу: https://3dnews.ru/939245.//ru.wikipedia.org/wiki/GDDR5.

Отримано 29.06.2017

УДК 004. 925.9

МЕТОД МНОГОПРОХОДНОГО РЕНДЕРИНГА ОБЪЕМНЫХ ДАННЫХ ДЛЯ МЕДИЦИНСКИХ ПРИЛОЖЕНИЙ

Романюк С. О.
Винницкий национальный технический университет
Вяткин С.И.
Институт автоматики и электрометрии СО РАН

Пропонується метод візуалізації на базі GPU-прискорювачів, який дозволяє відображати сцену, представлену кількома довільно розташованими об'ємами, пересічними з непрозорими або напівпрозорими геометричними об'єктами.

ВВЕДЕНИЕ

Визуализация нескольких наборов данных на основе измерений становится все более распространенной в современных научных приложениях. Это особенно важно в медицинских применениях. Например, в нейрохирургии мозга, невозможно визуализировать все интересующие структуры, используя только один метод медицинской визуализации. Различные методы должны быть объединены, чтобы иметь возможность видеть и анатомические структуры, и функциональные области. Одной из наиболее сложных компьютерных задач является отображение объемных (трехмерных) расчётных данных, или экспериментальных данных, полученных трехмерными методами сканирования. Методы сканирования, результатом которых являются объемные данные, включают магнитный резонанс (MRI), компьютерную томографию (CT), диффузионно-тензорную методику визуализации (ДТИ), ультразвук и т.д. Эти данные важно также совместить с дополнительными предметами, которые могут быть представлены геометрическими примитивами. Для нейрохирургических применений необходимо добавить виртуальные хирургические инструменты, опорные сетки и т.д. Кроме того, для изучения скрытых структур объемных данных, должна использоваться полупрозрачность, с использование передаточной функции прозрачности и поверхностного отсечения. Разные наборы данных, применяемые для одного и того же пациента часто имеют разное разрешение. В простейшем случае при объёмной визуализации необходимо обойти весь объем, рассматривая каждый воксель как 3D

точку, которая преобразуется в зависимости от положения наблюдателя, затем проецируется в Z-буфер и рисуется на экране. Такой способ визуализации называют рендерингом в объектном пространстве (object-space rendering) или рендерингом прямого хода (forward rendering). Необходимо отметить, что, в чистом виде, этот метод не обеспечивает визуализацию полупрозрачных материалов, так как воксели отображаются на экран в произвольном порядке. В простейшем случае при объёмной визуализации необходимо обойти весь объем, рассматривая каждый воксель как 3D точку, которая преобразуется в зависимости от положения наблюдателя, затем проецируется в Z-буфер и рисуется на экране. Такой способ визуализации называют рендерингом в объектном пространстве (object-space rendering) или рендерингом прямого хода (forward rendering) [4]. Необходимо отметить, что, в чистом виде, этот метод не обеспечивает визуализацию полупрозрачных материалов, так как воксели отображаются на экран в произвольном порядке. Расчет освещенности служит для определения цвета вокселя на основе некоторых классификационных и передаточных функций, а также моделей освещенности, которые варьируются от простейших (пороговое отсечение, за которым следует выборка цвета по таблице и закраска с учетом глубины) до наиболее сложных (сегментация с последующим применением специальной передаточной функции, учитывающей рассеивание) [1-3]. Алгоритм BTF (back-to-front) по существу совпадает с алгоритмом прямого хода, но отличается предварительной сортировкой массива вокселей, которая позволяет сканировать его компоненты в порядке уменьшения или увеличения расстояния до наблюдателя. Используя предварительно отсортированные массивы вокселей, в BTF-алгоритме обход объема производится в порядке уменьшения расстояния до наблюдателя, а Z-буфер для отсева невидимых вокселей становится не нужен. Алгоритм рисования просто записывает очередной воксель поверх уже отображенных или смешивает его со значением экрана, основываясь на вычислениях, моделирующих прохождение света через разные материалы. Алгоритм FTB (front-to-back) в основном совпадает с BTF, однако в нем воксели обходятся в порядке возрастания расстояния. Таким образом, можно легко реализовать визуализацию полупрозрачных объектов и в BTF, и в FTB, поскольку в этих методах объекты отображаются на экран в том же порядке, в котором по сцене проходят световые лучи [1-3].

Предлагается задачу разбить на две части, в первой – рэй кастинг, позволяющий в интерактивном режиме, получать высокое

качество рендеринга и перспективные проекции; и во-второй – сортировка по глубине, которая позволяет естественно интегрировать объемы, с непрозрачными и полупрозрачными геометрическими поверхностями (например, для визуализации хирургических инструментов).

ОПИСАНИЕ МЕТОДА

Выборка нескольких объемов независимо за один проход проблематично, особенно, если они произвольно пересекаются и полупрозрачные. Корректно вычислить цвет и прозрачность вдоль каждого луча можно только на каждом шаге выборки с учетом всех поверхностей. Предлагается многопроходный подход с использованием сортировки по глубине. Для этого сцена делится на области, в которых условия выборки не меняются. Рендеринг объемов и поверхностей происходит итерационно. Используется многопроходная техника на уровне фрагментов. Для сортировки по глубине используется OpenGL Framebuffer расширение и программируемые шейдеры фрагментов для выделения различных слоев и хранения их в памяти текстур. Рэй кастинг начинается с фронтальных граней в сцене с сохранением значений глубины и цвета фрагментов. Все слои проходят тест по глубине. Во втором проходе сцена растрируется снова. Композитные цвет и прозрачность накапливаются в буфере, который используется для нового прохода. Это позволяет отображать вместе с воксельными данными другие объекты (например, встраивать указатели, протезы, инструменты и т.д.), созданные приложением с помощью OpenGL или Direct3D. Но, так как для этого требуется несколько проходов по сцене, это снижает производительность. Используются 8-, 16-, и 32-битные данные, которые могут содержать до 4 полей, классификация содержимого которых (т.е. процесс преобразования значений в цвет и прозрачность) может независимо настраиваться, как и функции смешивания этих цветов и прозрачностей в процессе получения финального изображения. Например, одно поле может содержать информацию о плотности, а другое о типе вещества, тогда мы можем брать прозрачность для отображения из первого поля, а окрашивать, сообразуясь с типом вещества, заданным вторым полем. Или данные могут иметь 4 поля, содержащие цвета и прозрачности и отображаться. Можно сначала классифицировать данные, а потом интерполировать получившиеся цвета и прозрачности или наоборот – получать цвет и прозрачность по предварительно интерполированным данным. При этом новые данные могут загружаться, пока прежние

отображаются акселератором [5] Поддерживается до 10 секущих плоскостей с задаваемой толщиной. Фильтрация по прозрачности, значению или направлению градиента данных позволяет выделять поверхности и слои. Можно использовать в качестве градиентов при визуализации как вычисленные акселератором значения, так и непосредственно значения какого-либо поля, т.е. качественно восстанавливать поверхности по градиентным полям, если информация о величине градиента передаётся прямо в одном из полей воксела. Механизм «раннего прекращения обработки лучей» (early-ray termination) позволяет, при достижении некоторого порогового значения прозрачности прерывать дальнейший расчёт для лучей, уменьшая время обработки изображения.

ЗАКЛЮЧЕНИЕ

Представлен метод визуализации множественных, частично перекрываемых т пересекающихся объемов с произвольным числом непрозрачных и полупрозрачных геометрических фигур. Метод подходит для применения в нейрохирургии, где необходимы наборы данных с различными разрешениями, с геометрическими данными и дополнительными моделями поверхностей.

ЛИТЕРАТУРА

1. Drebin, R.A., Carpenter, L., and Hanrahan, P., "Volume Rendering", //Computer Graphics, 22(4):65-74, August 1988.
2. Sobeirajski L., D. Cohen, A. Kaufman, R. Yagel, and D. Acker, "A Fast Display Method for Volumetric Data", //The Visual Computer, 10(2):116-124, 1993.
3. Cabral B., N. Cam and J. Foran, "Accelerated Volume Rendering and Tomographic Reconstruction Using Texture Mapping Hardware", //Proceedings 1994 Symposium on Volume Visualization, pp.91-98.
4. Романюк О. Н. Високопродуктивні методи та засоби зафарбовування тривимірних графічних об'єктів. Монографія. / О. Н. Романюк, А. В. Чорний. - Вінниця : УНІВЕСУМ-Вінниця, 2006. — 190 с.
5. Романюк О.Н. Класифікація графічних відеоадаптерів / О. Н. Романюк, Р. Ю. Довгалюк, С. В. Олійник // Наукові праці Донецького національного технічного університету .Серія Інформатика, кібернетика та обчислювальна техніка". Випуск 14(188), — 2011. С. — 211-215.

Отримано 26.06.2017

УДК 004. 925.9

DEFORMABLE INTERACTIVE VOLUME-ORIENTED RENDERING BY PERTURBATION FUNCTIONS

A. N. Romanyuk
Vinnitsa National Technical University
S. I. Vyatkin,
Institute of Automation and Electrometry SBR

Відтворення об'єктів з рельєфною поверхнею на основі функцій в даний час потребує реалізації окремих процесів для деформації та візуалізації. Пропонується метод об'єднання цих процесів без побудови проміжних деформованих багатокутних поверхонь.

INTRODUCTION

The possibility of interactive generation of shapes of three-dimensional objects was considered in [1–3]. Before these publications, no reliable methods for modeling functional objects were developed. The objects were described manually in a text form. This operation is extremely labor-consuming. Therefore, it became very important to have a suitable source environment for editing three-dimensional objects. Systems for modeling polygonal objects were first developed. The AutoCAD computer-aided design system (version R11) offered a possibility of three-dimensional simulations. As an alternative for this system of computer-aided engineering, a bCAD system was developed, which provides both two-dimensional and three-dimensional design technologies and built-in visualization tools.

More recent developments include various commercial systems, such as SoftImage, Alias Wavefront (Maya), 3D Studio Max, LightWave 3D, Modo 3D software system for three-dimensional modeling and rendering, etc. There are interactive systems for modeling on the basis of polynomials, implicit surfaces [4], triangular grids [5], images [6], volumes [7], and functionally defined models [8, 9].

For polygonal models, it is difficult to ensure transformations of geometric objects with the use of operations over functions, which is necessary to perform unary and binary geometric operations and to detect collisions between the objects.

In the case of polygonal definition of the objects, it is rather difficult to compute surface deformation because each apex of the triangular grid has to be subjected to geometric calculations.

Today's realistic graphics requires a compact definition of a scene with an accurate description of threedimensional objects, which ensures effective implementation of various geometric operations with object models in simulations of the behavior of interacting objects.

Thus, the low level of realism of polygonal models and limited functional capabilities in terms of formation of graphical scenes give rise to an academic and simultaneously applied problem. Solving this problem requires the development of theoretical grounds for modeling and visualization of three-dimensional objects of a virtual medium, which would eliminate the drawbacks typical for the polygonal definition of object models.

Three basic types of functions are usually used for modeling object shapes: parametric, explicit, and implicit functions. Parametric functions are included into many modeling systems, in particular, the NURBS function, which is a mathematical form (particular case of B-splines), which is used in computer graphics for generation and presentation of curves and surfaces. Examples of explicit and implicit functions are soft objects [10], droplet model [11], and perturbation functions [12]. However, it is often rather difficult to calculate the defining function at a given point in functional definitions of surfaces. For this reason, preliminary triangulation of the surface is often performed for interactive modeling [9].

Available methods of object definition on the basis of functions are limited to a rather narrow group of modeled surfaces and slow visualization. One of the main drawbacks of available visualization methods is the difficulty in calculating surface points. Some methods do not guarantee finding surface points. The weak points of the known methods of definition of functional objects and visualization were described in more detail in [14].

In this paper, we consider an interactive modeling approach, which is a tool for creating three-dimensional objects without preliminary triangulation. A method that involves using graphics processing units is used for rapid visualization [14]. A source environment is developed, which allows interactive generation or editing of three-dimensional objects with the use of perturbation functions. The proposed software tools significantly simplify generation of functionally defined models.

OBJECTS DEFINED ON THE BASIS OF PERTURBATION FUNCTIONS

Functionally defined objects are constructed from second-order surfaces with analytical perturbation functions, which ensures reaching a high coefficient of geometric compression of highly realistic three-dimensional objects. It is proposed to describe geometric objects (free

shapes) by defining the function of deviation (of the second order) from the basic surface of the second order (quadrics) [12].

The function is defined by a second-order algebraic inequality with three unknowns x, y, z in the form $F(X) \geq 0$. The surfaces are considered as closed subsets of the Euclidean space E3, which are defined by the describing function $F(X) \geq 0$, where F is a continuous real function and X = (x1, x2, x3) is a point defined by coordinate variables in the space E3. The expression $F(X) > 0$ defines the points inside the surface, $F(X) = 0$ defines the points on the boundary, and $F(X) < 0$ defines the points located outside and not belonging to the surface.

Free shapes are constructed on the basis of quadrics. A free shape is a composition of the basic quadrics and perturbations. The surface is smooth if the exponent of the disturbing quadric is greater than two. This condition guarantees the continuity of the function and its derivative. There are several basic types of quadrics and perturbation functions [12].

The proposed method of describing objects of three-dimensional scenes by basic surfaces and functions has a compact presentation, which allows one to reduce the volume of data transfer by a factor of 10 to 1000, depending on particular three-dimensional scenes and models. Objects with flat faces can be also defined fairly easily, e.g., a cube can be defined by three quadrics. Moreover, in solving the describing function in the form of the inequality $F(X) \geq 0$, it is possible to visualize not only the surface, but also the internal structure of the object.

METHOD OF INTERACTIVE DEFORMABLE VOLUME-ORIENTED RENDERING

To create a complex scene, it is necessary to describe a certain number of primitives necessary for a particular problem. The entire three-dimensional scene is presented by an object, which is contacted by the rasterization algorithm by means of queries. Therefore, the geometric model should allow one to design objects and their compositions of arbitrary complexity. This is primarily reached by using Boolean operations of union and intersection. The entire scene has the form of a tree, each node of this tree is a design object, which performs logical operations with its descendants, and the top of the tree consists of primitives used by the system. When the rasterization algorithm addresses the design object with a certain query, this object asks his descendants, transforms the result, and provides an appropriate answer to the query. The descendant can be a primitive or another design object. If geometric operations, rotations, displacements, or scaling is applied to the design object, it repeats all these operations with his descendants.

Thus, the scene is described by a binary tree whose root is the object representing the entire threedimensional scene and possessing a

property of answering a query about its intersection with a volume. The nodes of the tree contain unite or intersection operators. The operator is also an object, while the leaves are objects expressed by quadrics.

The library of classes of functionally defined objects has a static chart for names of objects registered in the library and for creating functions. The names of the registered objects are automatically defined as lexical units in the syntax analysis of the scene. The goal of the preparatory phase is to construct a geometric model. The input data for the code are several text files, and each file describes a logically self-consistent element of the scene and all necessary parameters of this element. A pre-processor translates these files into one file containing a complete description of the scene. The resultant file is transferred to the input of the syntax analyzer (parser), which analyzes the text structures and constructs the model. When the model is successfully constructed, it is subjected to a projective transformation, which finalizes the pre-processing phase.

The problem of object design reduces to the problem of deformation of the basic surface in a required manner rather than to its approximation by primitives. This process resembles playdough sculpturing with the use of geometric operations described in [15]. The coordinates of the points of the deformed surface are determined by the method of detection of collisions of functionally defined objects [16]. This method of collision detection is based on the relation of intersection of the objects and recursive division of the object space for searching for the first point of the contact between the objects.

Deformation implies a possibility of adding a perturbation with parameters defined by the tool to an arbitrary point on the surface. The tool is chosen by the user (e.g., a thin cylinder) and defines the perturbation type and the domain of its action. For this purpose, a library of objects (tools) is created, i.e., a perturbation with prescribed parameters can be added to an arbitrary point of the surface by using a chosen tool. There is also a function of object extraction from the scene, aimed at performing certain operations with the object: rotation, compression–extension, motion, etc. Thus, the perturbation is added onto the object surface at a place indicated by the mouse. As it was already mentioned, the data are stored in the

memory in the form of a tree, where each object of the scene is presented as a description of the function with its properties. The tree is shown on the screen; it is possible to copy, cut, and paste individual branches or the tree as a whole. It is also possible to open a window of properties for each element of the tree and change the element properties in this window. The actions in the main window of the code are performed over the belongs to the object and for adding the depth levels

to the object and references to the parent, the code has a flag, which signalizes subtree participation in rendering. After that, the reference to the object is stored if this object has no descendants. Thus, the tree roots are found. A field in the class VxObject is created, which indicates the tree depth. The subtree depth is calculated directly before finding the minimum common parent for all objects in the list. If the depth is calculated during scene construction, it may become necessary to recalculate the depths of all nodes after the unite or intersection operations, which requires a lot of time; for this reason, only the depths of the roots are calculated. An important issue is the possibility of deformation of the initial shape, not only of adding new perturbations.

One part of the interacting system is the VxDemo code designed for visualization of unctionally defined objects. The input data for VxDemo are contained in a text file (with the SCN extension) created in some text editor or interactively. The file contains the scene description. The text of the file contains a set of lexical units, which can be conventionally divided into the following categories: graphical primitives, properties of primitives, operators, texture, and illumination parameters.

The output data are the resultant images of the scenes in the graphical format. The parameters of scene visualization can be changed in the Options window, which is opened by using the corresponding item in the View window.

The tool defines the perturbation type and the domain of its action. For this purpose, the user is offered certain graphical information (including extraction of the object, e.g., by a certain color or by a bounding box, drawing of axes, etc.). All drawings supporting user's working with operations are performed by OpenGL for obtaining a higher speed and functionality.

Thus, the main specific features of the rendering class are extraction into a separate thread, OpenGL support (compatibility of buffers, depth testing. etc.), and also the formation of bounding boxes for the objects.

An individual thread is necessary to ensure a possibility of interrupting and restarting of rendering after any change in the scene performed by the user.

The following problems were solved: response of the code to events indicated by the mouse (choosing and modifying the objects), extraction of the object (or several objects) in the scene and possibility for the user to perform some operation with the object(s); choosing the operation — affine transformations MRS (move, rotate, scale), geometric operations, and deformation, possibility of working with the list of tools, recording into a file and loading of the tree of the scene, and possibility of creating new tags and their recognition.

The code implementing the method of interactive modeling of functionally defined objects is written in the high-level language C++. This language is well suitable for the object model of the described system because it is an object-oriented language.

Necessary methods and classes of C++ were developed and implemented: the method of binary search for image elements, classes of functionally defined objects, classes of rendering of functionally defined objects, and classes of the interface of the system of interactive volume-oriented geometric modeling with a possibility of implementing a simple mechanism of extension of new algorithms and characteristics.

These hierarchical classes compose the core of the computational system, which can be extended to add other functions or to change the characteristics.

CONCLUSIONS

Many systems for interactive modeling and editing of functionally defined shapes involve intermediate triangulation of the edited surface for the purpose of decreasing the computation time, which complicates the modeling method itself and the entire system as a whole. Moreover, the computation accuracy is deteriorated by these intermediate manipulations. It should be also noted that successful triangulation is not possible for all geometric shapes; hence, there are constraints on the complexity of modeled shapes. In the proposed approach, we managed to avoid additional triangulation operation, while the interactive regime is retained. This is ensured by using appropriate methods of definition of three-dimensional objects, visualization, detection of object collisions, geometric operations, and possibility of local and global deformation.

As a result, a source environment that allows interactive formation and/or editing of functionally defined objects and a software system that significantly simplifies construction of functionally defined objects with the use of perturbation functions were developed.

REFERENCES

1. R. Szeliski and D. Tonnesen, "Surface Modeling with Oriented Particle Systems," in Proc. of the 19th Ann. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH'92) (ACM, New York, USA, 1992, pp. 185–194).

2. W. Welch and A. Witkin, "Free-Form Shape Design Using Triangulated Surfaces," in Proc. of the 21st Ann. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH'94) (ACM, New York, USA, 1994, pp. 247–256).

3. B. Curless and M. Levoy, "A Volumetric Method for Building Complex Models from Range Images," in Proc. of the 23rd Ann. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH'96) (ACM, New York, USA, 1996, pp. 303–312).

4. R. N. Perry and S. F. Frisken, "Kizamu: a System for Sculpting Digital Characters," in Proc. of the 28th Ann. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH'01) (ACM, New York, USA, 2001, pp. 47–56).

5. M. Agrawala, A. C. Beers, and M. Levoy, "3D Painting on Scanned Surfaces," in Proc. Symp. on Interactive 3D Graphics (I3D'95) (ACM, New York, USA, 1995, pp. 145–150).

6. B. M. Oh, M. Chen, J. Dorsey, and F. Durand, "Image-Based Modeling and Photo Editing," in Proc. of the 28th Ann. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH'01) (ACM, New York, USA, 2001, pp. 433–442).

7. E. Ferley, M.-P. Cani, and J.-D. Gascuel, "Virtual Sculpture," The Visual Computer 16 (8), 469–480 (2000).

8. A. Sourin, "Functionally Based Virtual Embossing," The Visual Computer 17 (7), 258–271 (2001).

9. K. Levinski and A. Sourin, "Interactive Function-Based Shape Modeling for Cyberworlds," in Proc. of the Intern. Conf. on Cyberworlds, (Tokyo, Japan, 18–20 November, 2004), pp. 54–61.

10. G. Wyvill, C. McPheeters, and B. Wyvill, "Data Structure for Soft Objects," The Visual Computer 2 (4), 227–234 (1988).

11. B. Wyvill, A. Guy, and E. Galin, "Extending the CSG Tree. Warping, Blending, and Boolean Operations in an Implicit Surface Modeling System," Computer Graphics Forum 18 (2), 8–24 (1999).

12. S. I. Vyatkin, "Complex Surface Modeling Using Perturbation Functions," Avtometriya 43 (3), 40–47 (2007)

[Optoelectron., Instrum., Data Process. 43 (3), 226–231 (2007)].

13. K. Levinski and A. Sourin, "Interactive Polygonisation for Function-Based Shape Modelling," Eurographics 2002, Saarbr¨ucken, Germany, September 2–6, 2002, pp. 71–79.

14. S. I. Vyatkin, "Method of Binary Search for Image Elements of Functionally Defined Objects Using Graphics Processing Units," Avtometriya 50 (6), 89–96 (2014) [Optoelectron., Instrum., Data Process. 50 (6), 606–612, (2014)].

15. S. I. Vyatkin, B. S. Dolgovesov, and A. T. Valetov, "Geometric Operations for Functionally Defined Objects Using Perturbation Functions," Avtometriya 40 (1), 65–73 (2004).

16. S. I. Vyatkin, B. S. Dolgovesov, and A. S. Korsun, "Detection of Collisions of Functionally Defined Objects in Problems of Computer Graphics," Avtometriya 39 (6), 119–126 (2003).

Received 20.06.2017

УДК 004. 925.9

МОДЕЛЮВАННЯ ГЕКСАГОНАЛЬНОГО РАСТРА НА КВАДРАТНОМУ РАСТРІ

Романюк О. Н., Мельник О.В., Стукач О.В.
Вінницький національний технічний університет

Розглянуто особливості гексагонального растра. Запропоновано методи емуляції гексагонального растра на квадратному растрі.

Гексагональний растр це растрова поверхня, базовим елементом якої (найменшим неподільним елементом) є гексагон.

Гексагон, або рівносторонній шестикутник – це двомірна геометрична фігура, що має шість рівних сторін, кути між сторонами якими рівні (рис. 1).

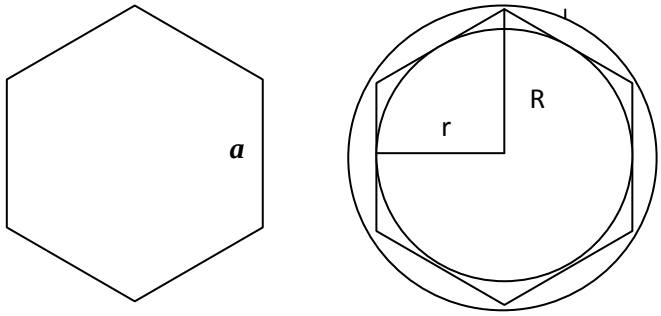


Рисунок 1 - Гексагон і вписане та описане кола.

Властивості правильного шестикутника (гексагона) - всі його сторони a рівні і дорівнюють радіусу описаного навколо шестикутника кола a = R. Всі кути гексагона дорівнюють 120°. Радіус вписаного кола $r = \frac{\sqrt{3}}{2}a$, найдовша діагональ правильного шестикутника вдвічі довша за його сторону.

Периметр гексагона $P = 6R = 4\sqrt{3}r$, а площа розраховується за формулою $S = \frac{3\sqrt{3}}{2}R^2 = 2\sqrt{3r^2}$.

Гексагон заданого розміру може замощувати будь-яку площину без розривів і накладань [1,2].

Замощена гексагоном площина (екрану) пристрою відображення графічної інформації називається гексагональним растром [1,2,3].

Оскільки гексагональний растр має подібність з існуючими в живій природі бджолиними стільниками, в багатьох авторів і дослідників використовується інша назва гексагонального растру – honeycomb, що в перекладі з англійської означає медові комірки (стільники) [2].

Гексагональний растр має рефлекційну симетрію, як і квадратний растр [4]. Але у гексагонального растра три лінії рефлекційної симетрії, в той час як в квадратного растра дві. Основні параметри гексагонального растру у порівнянні з квадратним растром наведено в таблиці 1.

Таблиця 1. Порівняння квадратного і гексагонального растру

Особливості	Тип растру	
	Квадратний	Гексагональний
Форма піксела	Квадрат	Гексагон
Кількість сусідніх пікселів	8	6
Мінімальна відстань до центра сусіднього піксела	1	1
Максимальна відстань до центра сусіднього піксела	1.414	1
Кількість ліній рефлекційної симетрії	2	3
Кут між осями координат (в градусах)	90	60

Можливо змоделювати гексагональний растр на квадратному растрі [5,6,7,8]. Для цього потрібно розрахувати систему координат, яка буде використана для адресації пікселів на растрі. Така система координат обов'язково повинна мати зазначені властивості: повнота - представлення будь-якого піксела на екрані в координатах; унікальність – одна пара координат для одного піксела; перетворюваність в декартову систему координат.

Одним із способів змоделювати гексагональний растр на квадратному – це збільшити в адресації пікселів координату x на значення ½ додавши до кожної координати кожного піксела x+1/2 (рис. 2).

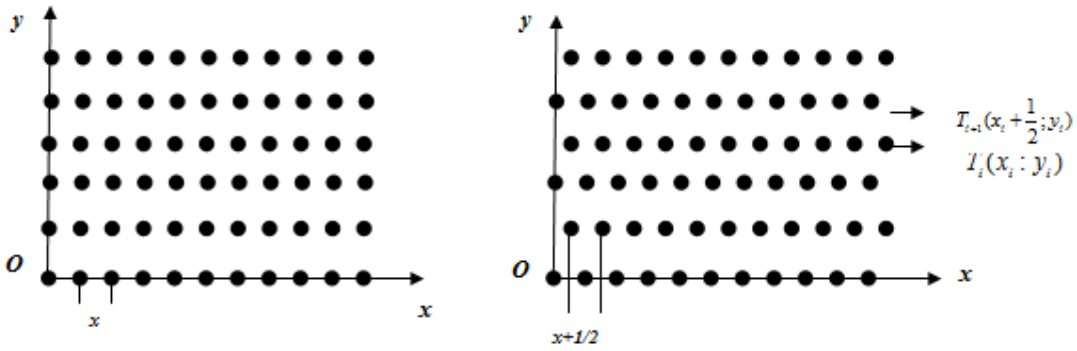


Рисунок 2 - Зміна адресації пікселів

Таким же способом можна зробити зсув по координаті у (рис. 3).

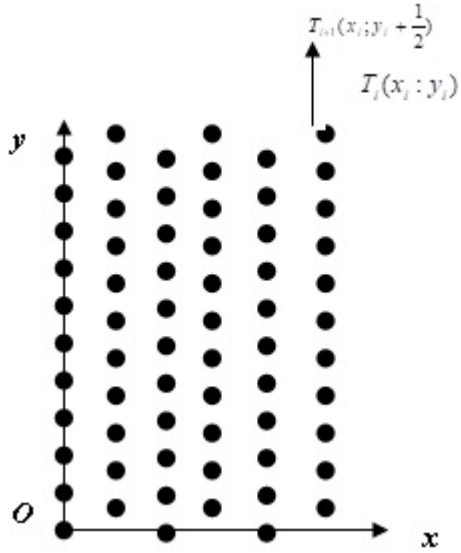


Рисунок 3 - Зміна адресації пікселів по осі у.

Таким чином змодельований псевдогексагональний растр дасть змогу застосувати алгоритми, розраховані для гексагональної моделі піксела і для гексагонального растра [1].

Ще оди із способів моделювання гексагонального растру на квадратному - не використовувати частину пікселів. Наприклад, через один ряд не використовувати всі непарні пікселі (рис.4).

Пікселі, що використовуються повинні мати збільшену інтенсивність кольору, оскільки збільшено порівняно зі звичайним растром відстань між пікселами.

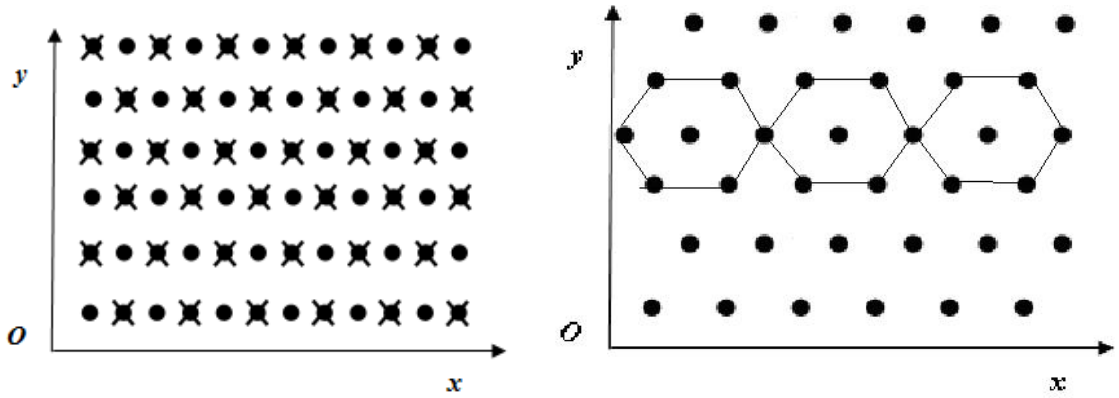


Рисунок 4 - Розміщення точок

Також можливе формування зображень на збільшеному в декілька разів растрі з подальшим використанням суперсемплінгу.

ЛІТЕРАТУРА:

1. Романюк О. Н. Особливості гексагональної моделі піксела / О. Н. Романюк, О. В. Мельник, // Міжнародний науково-технічний журнал «Вимірювальна та обчислювальна техніка в технологічних процесах», Хмельницький, ХНУ, 2014р. №1 (46) – 214 с. С. 91-95.

2. T. C. Hales, "The Honeycomb Conjecture,"/ Hales T. C. // Discrete Computational Geometry. Vol. 25. 2001. P. 1 – 22.

3. R. C. Staunton and N. Storey, "A comparison between square and hexagonal sampling methods for pipeline image processing," Proc. SPIE, vol. 1194. 1989. P. 142 – 151.

4. Reflection and rotation symmetry / Cambridge University Press. - [Електронний ресурс]; Режим доступу :
http://assets.cambridge.org/97805216/94315/excerpt/9780521694315_excerpt.pdf

5. Bell S. A digital geometry for hexagonal pixels / S. Bell, H. Fred, D. Mason // Image and Vision Computing. – V. 7. – No. 3. – 1989. - P. 194 - 204.

6. Luczak E. Distance on a hexagonal grid / E. Luczak, A. Rosenfeld // IEEE Transactions on Computers. – V.C - 25. – No. 5. – 1976. – P. 532 - 533.

7. Carstens B. Hexagonal domain transform for shape analysis / B. Carstens, M. Quinn // Intelligent Robots and Computer Vision X: Algorithms and Techniques, SPIE. – 1991. – V. 1607. – P. 197 - 205.

8. Wuthrich C. A. An Algorithmic Comparison Between Square and Hexagonal-based Grid / C. A. Wuthrich, P. Stucki // CVGIP: Graphical Models and Image Processing. – Vol. 53 – 1999. P. 324 - 339.

9. Романюк О.Н. Класификація методів антиаліайзингу / О. Н. Романюк, О.В. Мельник, С.І. Вяткін // Вістник ХНТУ, № 3(50), 2014 г –С 124-129.

Отримано 29.06.2017

УДК 004.4

СТВОРЕННЯ ІГРОВОГО ДОДАТКУ ЗАСОБАМИ ГРАФІЧНОГО СЕРЕДОВИЩА UNITY 3D

Тихонова О.А., Скрипник Т.В.
Донецький національний технічний університет

Наводиться приклад вирішення проблеми розробки і створення ігрового додатку квестового типу за допомогою програмного графічного середовища Unity 3D.

Ключові слова: Unity 3D, ігровий додаток, сценарій, ігрове меню, біом, скрипти, UML діаграма

1. ВСТУП.

У наш час складно уявити сферу створення програмних продуктів без графічних систем, зокрема, ігрових додатків. У зв'язку з цим стали популярні не тільки ігрові програми як такі, але й програмні середовища для спрощення виробництва систем даного типу. Існує чимала кількість таких середовищ, одним з найпопулярніших є Unity 3D, за допомогою якого і було реалізовано описаний ігровий додаток. Допоміжним середовищем для розробки ігрових об'єктів став, також відомий на ринку, 3ds Max, який пропонує широкий функціонал маніпуляції 3D моделями.

Актуальність даного напрямку росте з кожним роком, як і вік зацікавленої в розвитку даної галузі аудиторії. І частка ігор, спроектованих за допомогою Unity 3D, не залишає сумнівів в подальшому зростанні.

Як правило такі ігрові додатки є результатом командної роботи.

2. ПОСТАНОВКА ЗАДАЧІ

2.1. Узагальнений список поставлених завдань

1) Створити сценарій ігрового процесу
2) Створити ігрові об'єкти
3) Створити ігрові скрипти
4) Створити анімації до ігрових об’єктів
5) Створити ігрове меню
6) Створити сцену ігрового процесу
7) Реалізувати логіку ігрового процесу
8) Скомпілювати в додаток, що виконується

2.2. Технічні вимоги до ігрового додатку

1) Ознайомитися з інструментами для роботи в середовищі 3ds Max
2) Ознайомитися з інструментами для роботи в середовищі Unity 3D
3) Створити необхідні графічні об'єкти
4) Створити анімації для ігрових об'єктів
5) Створити програмні скрипти логіки роботи ігрових об'єктів
6) Налаштувати взаємодію графічних об'єктів з анімацією
7) Налаштувати взаємодію графічних об'єктів зі скриптами
8) Створити скрипти взаємодії ігрових об'єктів
9) Налаштувати взаємодію між ігровими об'єктами
10) Створити сцену меню
11) Створити сцену ігрового процесу
12) Розмістити ігрові об'єкти на сценах
13) Скомпілювати гру в файл, що виконується

2.3. Нетехнічні вимоги до ігрового додатку

1. Розробити сценарій ігрового процесу
2. Провести аналіз необхідних графічних об'єктів
3. Розробити логіку ігрового процесу
4. Розподілити обов'язки в команді

3. СТВОРЕННЯ СЦЕНАРІЮ ІГРОВОГО ПРОЦЕСУ

3.1 Ігрові етапи

В результаті аналізу було створено сценарій, що складається з 9 пунктів ("Старт", "Печера Python", "Дуб Java", "Кладовище Flash", "Пустеля Perl", "Лабіринт C ++", "Гора Ruby", "Вежа Swift "," Фініш ").

До цього списку варто додати пункт поза ітераційної залежності, який можливий, на будь-якому етапі ігрового процесу - пункт "Смерть".

Оскільки ігровий процес є поетапним, розумно буде зобразити його за допомогою UML діаграми (рисунок 1).

3.2 Діаграма сценарію ігрового процесу

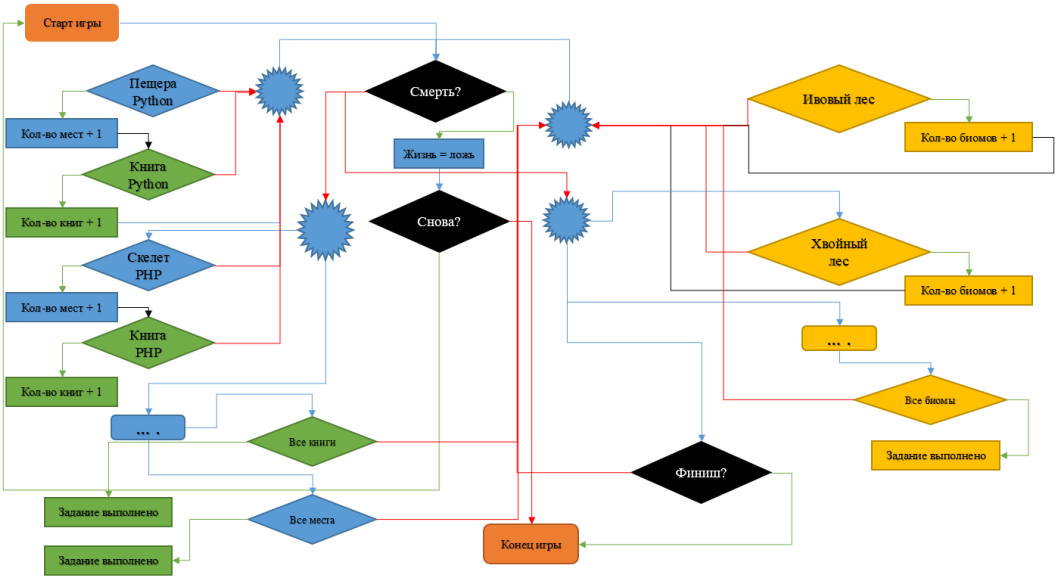


Рисунок 1 - Діаграма сценарію ігрового процесу

З цієї діаграмою стає ясно, що ігровий процес починається після запуску програми і закінчується закриттям програми. Обидві ці події контролюються з ігрового меню. Всі ігрові події, які зображені між «Старт гри» і «Кінець гри» контролюються на основний ігровий сцені і передбачають взаємодію гравця з ігровими об'єктами (див. рисунок 2).



Рисунок 2 – Ігровий маршрут

3.3 Ігрові події

- Збір певних ігрових об'єктів
- Знаходження гравця в певних координатах
- Необхідна кількість зібраних об'єктів

3.4 Концепт гри

Концепт гри полягає в досягненні ключового місця і зборі предметів на ігровій сцені з рішенням нескладних головоломок для їх пошуку.

Ігровий простір представлено у вигляді світу казок, гірська місцевість з шістьма біомами: (Річка, Пустеля, Хвойний ліс, Листяний ліс, Луг, Вербовий ліс, Водоспад, Мертвий ліс, Гори). Присутні споруди різного розміру, намети, багаття, покажчики і багато іншого. Приклади фрагментів ігрового простору зображені на рисунках 3 – 6:



Рисунок 3 – Печера Python

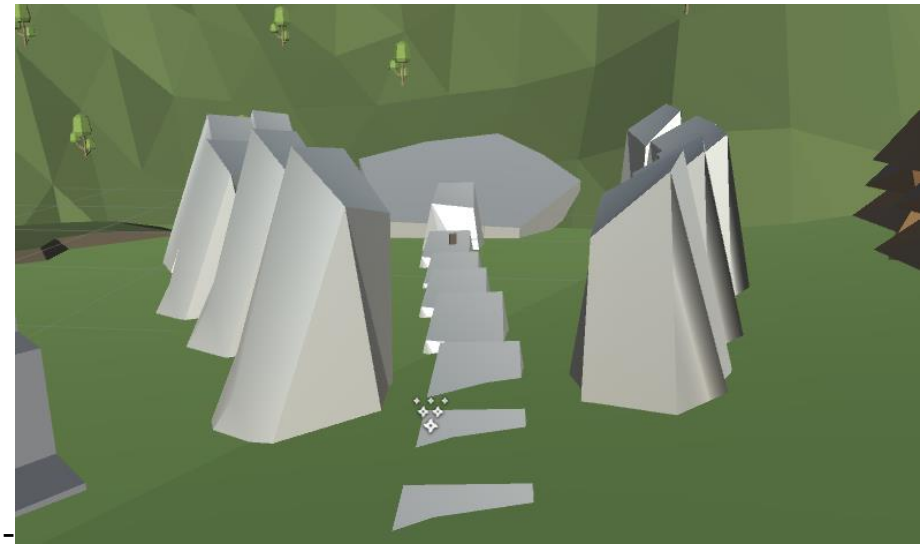


Рисунок 4 – Скелет PHP



Рисунок 5 – Дуб Java

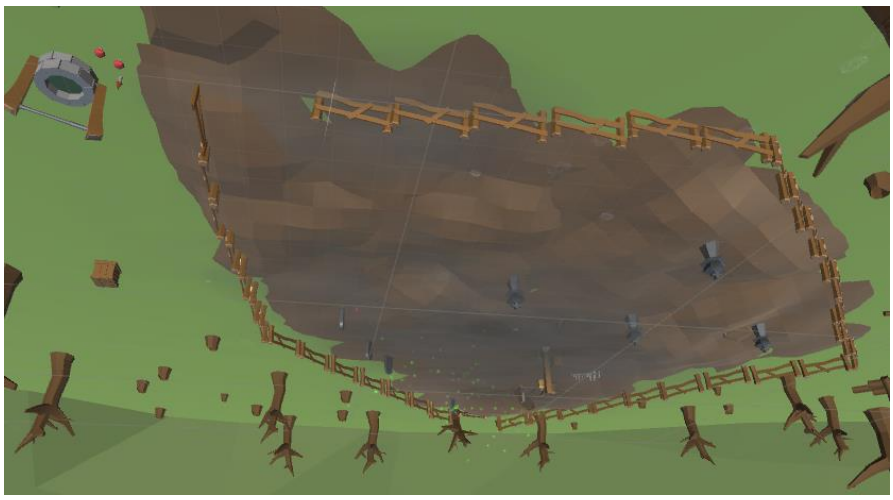


Рисунок 6 – Кладовище Flash

З об'єктів (явищ) представлено:

- Зміна дня і ночі
- Туман
- Хмари
- Дощ
- Вогонь
- Вибухи
- Дим
- Гуркіт блискавок
- Піщані бурі
- Світляки
- Водоспад

Приклади реалізації явищ показано на рисунках 7 – 10:



Рисунок 7 – Піщані бурі

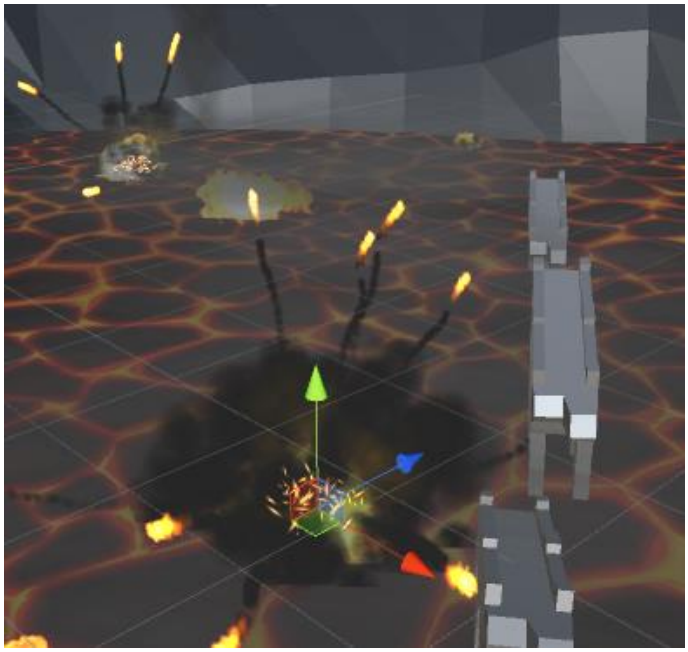


Рисунок 8 – Вибухи

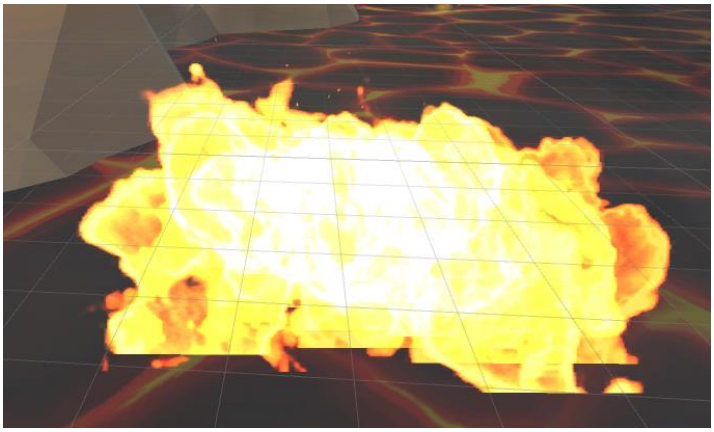


Рисунок 9 – Вогонь

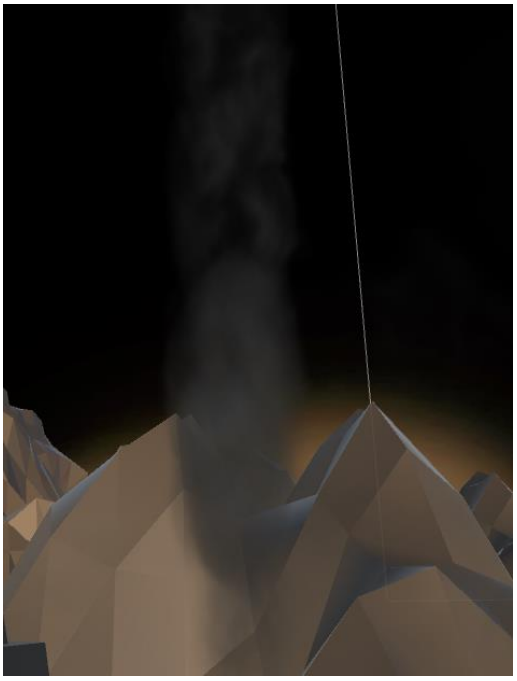


Рисунок 10 – Дим

Головний персонаж, гуманоїд європеоїдної раси, імовірно, чоловічої статі, вік невизначений, з метою більш сумісного уособлення гравця з персонажем. Його зображення проілюстровані на рисунку 11:



а) б) в)

Рисунок 11 – Вид та анімація головного героя.

4. РОЗРОБКА ДИЗАЙНУ РІВНЕЙ ГРИ

Дизайн рівнів - це важкий і кропіткий процес розстановки ігрових об'єктів, при цьому потрібно враховувати безліч чинників, таких як:

1) Пропорції предметів

Кожен предмет повинен мати вірні розміри в співвідношенні з іншими предметами і гравцем.

2) Доречність даного ігрового об'єкта в даному секторі

Побудова логічних ланцюжків появи даного предмета в даній ігровій області. Чи відповідає об'єкт вимогам ігрового всесвіту.

3) Налаштування різноманітності розмірів, форм і кутів повороту кожного ігрового об'єкта

Метою є саме різноманітність, створення видимості, випадкового / органічного розташування ігрових об'єктів на сцені.

4) Облік можливостей гравця

При побудові обмежень для гравця протягом усього ігрового маршруту враховується висота стрибка персонажа, можливість підніматися по нерівних поверхнях, тощо.

5) Насиченість ігрового процесу подіями

Необхідно рівномірно розподіляти навантаження подіями протягом усього ігрового процесу. Не давати гравцеві можливості занудьгувати, або розфокусувати увагу на ігрових завданнях.

6) Робота над «зануренням» гравця

Найціннішим, але в той же час і складним, є спроба створити відчуття «занурення» в ігровий процес, який характеризується співпереживанням, радістю перемог і сумом поразок, уособленням себе з головним героєм, тощо.

5. РОЗПОДІЛ ОБОВ'ЯЗКІВ У КОМАНДІ

Командна робота має на увазі одночасне виконання роботи за одиницю часу, що дозволяє скоротити його витрати.

З цією метою, було прийнято рішення об'єднати зусилля і тим самим збільшити масштаб і складність, а також поліпшити якість ігрового програми, за відведений час.

1) Учасники команди розділяють обсяг робіт, при цьому, беручи участь у всіх етапах проектування і реалізації проекту рівносильно.

2) Було проведено «брейнштормі», в яких генерувалися і обговорювалися ідеї реалізації вимог проекту.

3) Було спроектовано кілька версій ігровий логіки і дизайну, і по досягненні компромісу, було отримано підсумковий макет для реалізації.

4) У процесі тестування було виявлено, локалізовано і виправлено максимальну кількість дефектів, допущених в процесі розробки.

6. ВИСНОВКИ

Скомпільований ігровий додаток в середовищі Unity 3D, для функціонування якого потрібен персональний комп'ютер з операційною системою Windows і архітектурою процесора x86 або x64.

Як показує практика, моделювання, створення текстур і матеріалів, дизайн рівнів, написання сценарію і ігрової логіки, програмна реалізація ігрової логіки, командний менеджмент і тестування ігрового програми, це завдання абсолютно різних людей, або навіть груп людей, тому, як бути кваліфікованим фахівцем у всіх областях одночасно, це дуже рідкісна ситуація, якщо не неможлива. Крім рівня кваліфікації і навантаження, зростає ймовірність допущення великої кількості помилок всіх видів і класифікацій. Досить важливим є особистісний фактор і мотивація.

ЛІТЕРАТУРА

[1] Sue Blackman. Unity for Absolute Beginners [Електронний ресурс]. - Режим доступу:http://www.apress.com/la/book/9781430267799
[2] Janine Suvak. Learn Unity3D Programming with UnityScript [Електронний ресурс]. - Режим доступу: http://www.apress.com/us/book/9781430265863
[3] Alex Okita: Learning C# Programming with Unity3D [Електронний ресурс]. - Режим доступу: https://www.amazon.com/Learning-C-Programming-Unity-3D/dp/1466586524
[4] Cliff Bleszinski. The Art and Science of Level Design [Електронний ресурс]. - Режим доступу: http://leveldesign3r.blogspot.com/p/what-is-level-designer.html
[5] Обучающие материалы [Електронний ресурс]. - Режим доступу: https://unity3d.com/ru/learn/tutorials/

Отримано 25.06.2017

УДК 004.932.4

ДОСЛІДЖЕННЯ МЕТОДУ ДЕКОНВОЛЮЦІЇ ВІНЕРА ПРИ РЕАЛІЗАЦІЇ НА МОБІЛЬНИХ ПЛАТФОРМАХ

І.А. Філатов, С.О. Цололо
Донецький національний технічний університет, Україна

В роботі розглядаються особливості реалізації алгоритму деконволюції Вінера та створення фільтру Вінера для відновлення інформації з розмитих або розфокусуваних зображень. Виконано реалізацію і порівняльний аналіз декількох способів прискорення роботи алгоритму при його використанні на мобільних платформах.

Ключові слова: реконструкція зображень, деконволюція та фільтрація Вінера, дискретне перетворення Фур'є, швидке перетворення Фур'є

1. ВСТУП

Проблема відновлення зображення з'явилася в століття становлення цифрової фотографії. Змаз або розфокусування були відомі і раніше, але з появою цифрового представлення фотографії вже є не тільки проекцією відбитого світла через лінзу фотоапарата на плівку. Тепер фотографія – це дискретні відліки, які при наявності процесора можна обробити і перетворити. Фільтр Вінера з'явився задовго до появи цифрової фотографії, в 1942 р. в класичній праці «Екстраполяція, інтерполяція і згладжування часових рядів». За допомогою теорії оцінок він точно обчислив характеристики фільтра, який забезпечив найкраще виділення необхідного сигналу на тлі перешкод [1].

У наш час вже є розробленими методи відновлення зображень, на основі яких побудовано величезну кількість додатків, що так чи інакше оброблюють зображення. Але більшість цих додатків розроблені під конкретні системи (або сімейства систем), що не дозволяє повноцінно використовувати можливості цих програм на різних платформах.

Отже, метою роботи є дослідження *особливостей реалізації алгоритму деконволюції Вінера при реалізації у додатку* для платформи Android, аналізі швидкості і якості роботи алгоритму

Основні задачі, що вирішуються у роботі:

1. Аналіз алгоритму з метою виділення особливостей алгоритму, що можуть бути використані для збільшення швидкодії алгоритму.

2. Аналіз особливостей та припущень платформи Android, що можуть бути використані при реалізації алгоритму
3. Реалізація алгоритму деконволюції Вінера з урахуванням виявлених особливостей.
4. Дослідження швидкості при різних розмірах зображення та підходах до реалізації.

2. АЛГОРИТМ ДЕКОНВОЛЮЦІЇ ВІНЕРА

В основі методу деконволюції Вінера є розрахунок фільтра, за своїми властивостями схожим з інверсним фільтром, але менш схильним до впливу шуму на зображенні, і мінімізації його появи при розрахунках. Для роботи потрібно власне саме зображення (вже схильне до розмиття або розфокусування) і характеристики функції розтікання точок – знань про те, до яких змін схильний сигнал [2].

Зображенням будемо вважати картинку з 3 кольоровими шарами – червоним, синім і зеленим. Усі три шара схильні до розмиття або розфокусування однаково, тому фільтр для всіх них потрібен лише один. Після обробки цих – потоків інформації (зображення і функції розтікання точок – на виході отримаємо результуюче зображення такого ж розміру, як і вихідне, але з порівняно кращою якістю зображення. Схема алгоритму деконволюції Вінера наведена на рис. 1.



Рисунок 1 – Схема алгоритму деконволюції Вінера

При цьому слід виділити такі особливості:

1. Алгоритм покликаний працювати з матрицею значень, збільшення матриці функції розтікання точок до розмірів зображення буде описано нижче.
2. Циклічні зсуви потрібні для правильного позиціонування отриманої в попередньому кроці матриці і для подальшого перетворення цієї матриці в фільтр. Зсуви виконуються таким чином, щоб центр вихідної матриці функції розтікання точок лежав строго в лівому верхньому кутку.
3. Алгоритм працює з оптичною передавальною функцією, яку потрібно отримати з функції розтікання точок (рис. 2) [3].

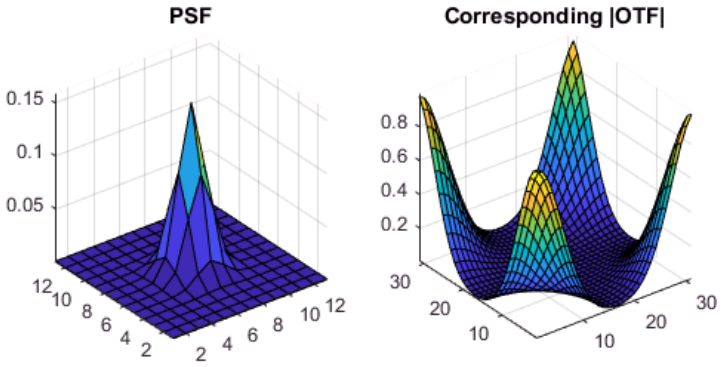


Рисунок 2 – Перетворення функції розтікання точок в оптичну передавальну функцію.

Слід зазначити, що на даному етапі виникають складності обчислення двовимірного дискретного перетворення Фур'є (ДПФ) від матриці. В загальному випадку рівняння ДПФ має вигляд (1):

$$\mathbf{X_p(k_1,k_2) = X(z_1,z_2)|}_{\mathbf{z_1}=e^{j\left(\frac{2\pi}{N_1}\right)k_1},\mathbf{z_2}=e^{j\left(\frac{2\pi}{N_2}\right)k_2}} = \sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1} x(n_1,n_2)e^{-j\left(\frac{2\pi}{N_1}\right)n_1k_1}e^{-j\left(\frac{2\pi}{N_2}\right)n_2k_2} \quad (1)$$

Після реалізації були проведені дослідження продуктивності методу, що виявило проблему вкрай малої швидкості виконання перетворення (рис.3). Така швидкість є закономірною для виробленого кількості обчислень, але для реалізації на мобільній платформі потрібно знайти більш швидкий спосіб розрахунку.

Одним з варіантів оптимізації алгоритму з точки зору часових показників є використання швидкого перетворення Фур'є. Ідея полягає в тому, що виконується проріджування сигналу по частоті, що дозволяє розділити вихідний сигнал на 2 рівних сигнали, кількість відліків кожного з яких є в 2 рази менше вихідного.



Рисунок 3 – Витрати часу на обчислення двовимірного ДПФ

Таким чином при максимальному проріджуванні буде досягнутий сигнал, що складається з двох відліків, і тоді перетворення можна записати у вигляді

$$y_k = y_k^0 + w_n^k y_k^1, k = 0 \ldots \frac{n}{2} - 1, \quad (2)$$

$$y_{k+n/2} = y_k^0 - w_n^k y_k^1, k = 0 \ldots \frac{n}{2} - 1.$$

При тестуванні швидкості роботи результат відрізняється на кілька порядків. Швидкість розрахунку зростає, але при цьому можливе проявлення похибки. Розмір похибки – не більше 0,25% в порівнянні з ДПФ дозволяє проігнорувати її з урахуванням отримання значного росту швидкодії.

Для подальшого прискорення може бути використана оптимізація ШПФ за допомогою методу Кулі-Тьюки (рис. 4) [6]. Варто відзначити, що вищевказані методи працюють тільки з матрицями, розміри яких кратні ступеням числа 2 (4x4, 8x8, 16x16 тощо), отже зображення також потрібно доповнити відліками до таких розмірів.

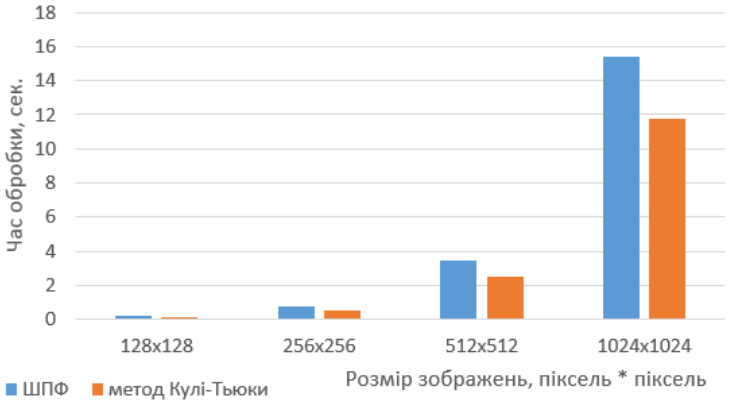


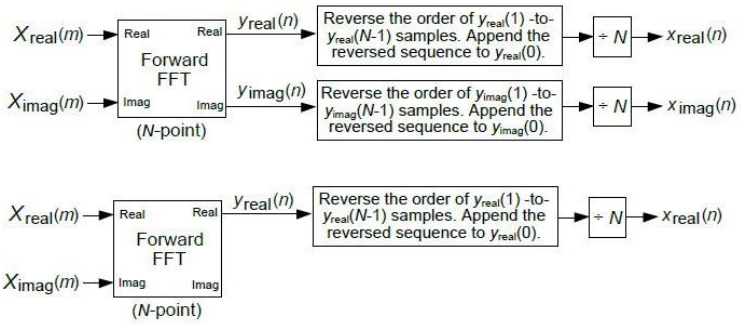
Рисунок 4 – Порівняння швидкості роботи методів ШПФ

3. РОЗРАХУНОК ФІЛЬТРУ ВІНЕРА

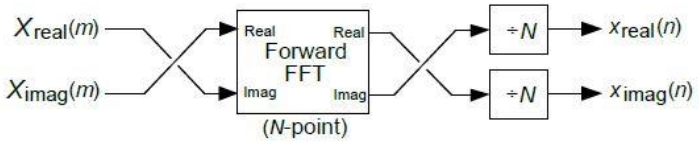
Для розрахунку фільтра необхідно виконати поелементне зведення матриці до другої степені, а потім розділити на перетворену матрицю. Будемо вважати, що зображення складається з раціональних позитивних чисел, при цьому в кожних 3 байтах картинки містяться послідовно значення червоної, зеленої і синьої компоненти, при цьому значення не можуть виходити за розміри байту (від 0 до 255).

Наступним кроком є зворотне перетворення Фур'є. Маючи готовий метод двовимірного ШПФ можна отримати зворотне за допомогою невеликих змін, є кілька способів обчислити зворотне ШПФ за допомогою ШПФ (рис. 5) [7].

а)



б)



в)

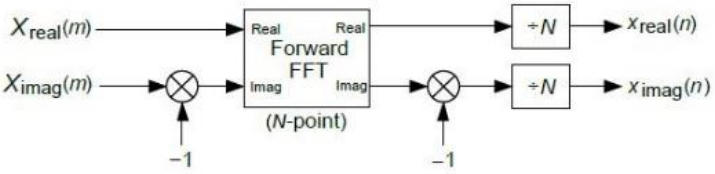


Рисунок 5 – Підходи до виконання зворотного ШПФ

Особливістю способу а) є те, що перед подачею на блок ШПФ або після перетворення відліки від 1 до N-1 зеркально змінюють позиції. У способі б) перед подачею на блок ШПФ та після перетворення сигнали обмінюються місцями.

Способи а) і б) більше підходять, коли в платформі реалізації присутній блок розрахунку ШПФ – тоді використовуючи здвиги в матрицях сигналу, що подається на вхід або виходить з блоку ШПФ, можна отримати правильний результат. Але саме для програмних способів реалізації є цікавим метод в), тому що він обумовлює найбільш просту реалізацію з використанням комплексно-сполучених

чисел. При цьому необхідно лише змінити знак -j на +j і розділити кожен елемент масиву на добуток розмірів масиву.

4. ВИСНОВКИ

В роботі було проведено аналіз алгоритму деконволюції Вінера, виявлені основі етапи та операції з попередньої обробки зображення та створенню фільтрів Вінера. Були проаналізовані способи реалізації деконволюції Вінера та створення фільтрів Вінера з точки зору їх ефективності для мобільних платформ. Для кожного із способів виконано реалізацію алгоритму, після чого проведений порівняльний аналіз швидкодії програми при різних початкових даних.

У подальшому для пришвидшення роботи планується використати засоби мови Java для розпаралелювання обчислень фільтруючих матриць та відфільтрованих зображень за допомогою пакету java.util.concurrent або інших допоміжних пакетів.

ЛІТЕРАТУРА

[1] Энциклопедия по машиностроению XXL. Фильтр Винера [електронний ресурс]. – Режим доступу: http://mash-xxl.info/info/215629/

[2] Point-spread function, spatial resolution and optical sectioning [електронний ресурс]. – Режим доступу: http://www.nature.com/nmeth/journal/v14/n4/box/nmeth.4230_BX1.html?foxtrotcallback=true

[3] Convert point-spread function to optical transfer function [електронний ресурс]. – Режим доступу: https://www.mathworks.com/help/images/ref/psf2otf.html

[4] Рабинер Л., Гоулд Б. Теория и применение цифровой обработки сигналов. – М: Мир, 1978. – 834 с.

[5] Быстрое преобразование Фурье за O (N log N). Применение к умножению двух полиномов или длинных чисел [електронний ресурс]. – Режим доступа: http://e-maxx.ru/algo/fft_multiply

[6] Липатов В.И. Реализация алгоритмов двумерного БПФ и сравнение их быстродействия в MathCad –Сибирский федеральный университет, 2005.

[7] Four Ways to Compute an Inverse FFT Using the Forward FFT Algorithm. Rick Lyons [електронний ресурс]. – Режим доступа: https://www.dsprelated.com/showarticle/800.php

Отримано 27.06.2017

УДК 004.932

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ЦИФРОВОЇ КОРЕКЦІЇ ТА ПІДВИЩЕННЯ ЯКОСТІ РАСТРОВИХ ЗОБРАЖЕНЬ

А.А. Яровий, Д.Г. Пасічник, Р.А. Василічишин
Вінницький національний технічний університет

Здійснено аналіз моделей цифрової корекції та підвищення якості растрових зображень. Визначено моделі, реалізацію яких ефективно адаптувати на основі технології GPGPU. Здійснено програмну реалізацію обраних моделей на основі CPU- та GPU-орієнтованої апаратної платформи, а також тестування розробленої інтелектуальної системи на базі зображень різної розмірності.

ВСТУП

Комп'ютерна графіка ефективно представляє образи реального світу. На відміну від людей, здатних сприймати електромагнітне випромінювання лише у видимому діапазоні, машинне оброблення зображень охоплює практично весь електромагнітний спектр від гамма-випромінювання до радіохвиль [1].

Методи цифрової корекції широко застосовуються в промисловості, мистецтві, медицині, космосі. Вони застосовуються при управлінні процесами, автоматизації виявлення і супроводу об'єктів, розпізнаванні образів і в багатьох інших додатках [1]. В даний час одним із актуальних напрямів інформатизації та моделювання є комп'ютеризація медичного обладнання, у зв'язку з чим растрова графіка відіграє важливу роль в медицині. У більшості випадків застосовують стандартне програмне забезпечення – офісні програми, графічні редактори, СУБД та ін. Тому розробка і впровадження інтелектуальних систем в області медичних технологій є досить актуальним завданням.

Метою даних досліджень є підвищення інформативності та якості растрових зображень шляхом їх цифрової корекції, а також швидкодії процесу їх оброблення на основі GPGPU-технологій.

АНАЛІЗ МОДЕЛЕЙ ЦИФРОВОЇ КОРЕКЦІЇ ТА ПІДВИЩЕННЯ ЯКОСТІ РАСТРОВИХ ЗОБРАЖЕНЬ

Існуючі підходи до підвищення якості цифрового зображення і відновлення його структури, загалом, поділяють на дві категорії:

оброблення в просторовій області (просторові методи), засновані на прямому маніпулюванні пікселями зображення, та оброблення в частотній області (частотні методи), засновані на модифікації (фільтрації) сигналу [1, 2].

Одним із розглянутих у даному дослідженні просторових методів підвищення якості зображень є еквалізація (вирівнювання) гістограми зображення.

На першому етапі відбувається побудова гістограми яскравості. Гістограми будують як для кольорових зображень по кожному з каналів, так і для зображень у grayscale моделі. Гістограма є графіком розподілу півтонів зображення, в якому по горизонтальній осі представлена яскравість, а по вертикалі – відносна кількість пікселів з даними значеннями яскравості.

На другому етапі виконується нелінійне перетворення, що забезпечує необхідні властивості вихідного зображення. При цьому замість невідомого істинного інтегрального розподілу використовується його оцінка, основана на гістограмі. З урахуванням цього, всі методи поелементного перетворення зображень, метою яких є видозміна законів розподілу, відносять до гістограмних методів. Зокрема, перетворення, при якому вихідне зображення має рівномірний розподіл, називається еквалізацією (вирівнюванням) гістограм [1, 2].

Результати моделювання даного методу оброблення зображень наведено на рис. 1.



а) б)

Рисунок 1 – Результати роботи алгоритму еквалізації гістограми зображення,
а) вхідне зображення, б) зображення із вирівняною яскравістю

Іншим розглянутим у даному дослідженні методом підвищення якості зображення є Retinex. Його алгоритм роботи стосується вирівнювання освітлення на зображенні та полягає в наступному. Саме

зображення формується як добуток низьких і високих частот, тобто самого освітлення і об'єкта за виразом [3]:

$$l(i,j) = G * I(i,j), \quad (1)$$

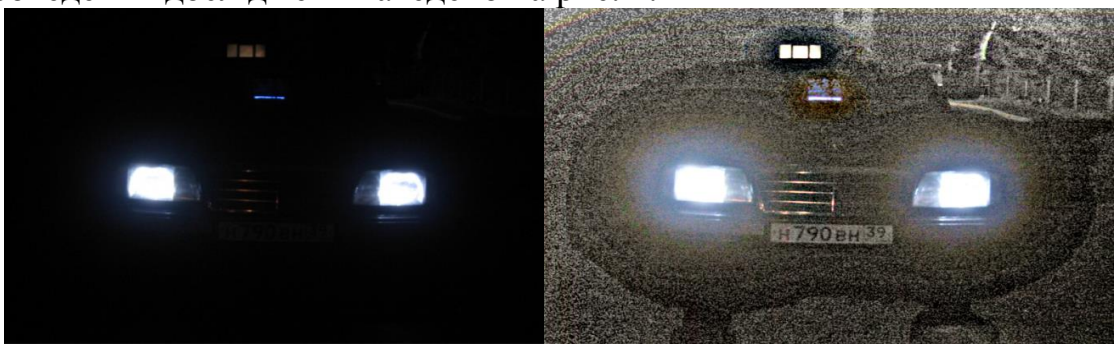
де l – освітленість; G – фільтр Гауса;I – сам об'єкт.

Відновлюється зображення за виразом [2]:

$$I'(i,j) = \sum_k w_k \, log\, I(i,j) - \, log\, g_k(i,j) * I(i,j), \quad (2)$$

де w_k – вагові коефіцієнти.

Результати оброблення зображень за даною моделлю в межах проведених досліджень наведено на рис. 2.



а) б)

Рисунок 2 – Результати оброблення зображень алгоритмом Retinex, а) вхідне зображення, б) зображення із вирівняним освітленням

Ще одним розглянутим у даному дослідженні методом підвищення якості зображень є застосування фільтру для підвищення різкості зображення. Даний фільтр реалізується на основі ядра згортки. Елемент зображення отримує нове значення на основі групи елементів, що примикають до даного. Область примикання є квадратною матрицею, розмірність якої збігається з розміром обраного ядра згортки, і центром в оброблюваному елементі [1, 2, 4].

Для типових зображень в більшості випадків для якісної обробки зображення достатньо застосувати один із методів цифрової корекції

Однак при вирішенні прикладних завдань для досягнення прийнятних результатів може знадобитися застосування декількох методів корекції які доповнюють один одного.

Така ситуація випливає з того, що застосування одного алгоритму корекції зумовлює недостатню корекцію, або (частіше всього) навпаки, занадто сильну, в результаті чого відбувається погіршення якості зображення, коли важливі дрібні деталі можуть залишитися невидимими.

Так, у проведених експериментальних дослідженнях на базі рентгенівських знімків, найефективнішим виявився такий підхід –

зображення обробляється паралельно трьома методами: за допомогою операторів Собеля (Sobel operator) і Лапласа (Laplace operator) та детектором контурів Кенні (Detector Canny). Після чого зображення оброблені за допомогою операторів Собеля і Лапласа перемножуються (Multiplication of images) і отримане зображення додається до контурів отриманих детектором Кенні (Add image). Отримане в результаті даних операцій зображення накладається на вхідне і обробляється за допомогою гамма-корекції (Gamma correction) [5].

Із покращенням якості при збільшенні рівня деталізації зображення виникає проблема швидкодії їх оброблення, адже вони займають значну кількість пам'яті. В свою чергу це призводить до проблеми ефективного редагування таких зображень. Оскільки великорозмірні растрові зображення займають значні масиви пам'яті, то для забезпечення роботи функцій редагування таких зображень необхідні значні ресурси комп'ютерних систем. У зв'язку з цим перспективною для оброблення такого роду зображень є GPU-орієнтовані програмно-апаратні платформи.

У проведених дослідженнях при порівняні швидкодії роботи алгоритмів підвищення якості зображення на CPU- та GPU-орієнтованих платформах (рис. 3), було виявлено незначне прискорення (близько 0,01 – 0,02 мс) роботи алгоритмів на GPU-орієнтованій платформі при обробленні зображень розмірністю до 1600×1200 пікселів. Із збільшенням розмірності зображень швидкодія оброблення на GPU-орієнтованій платформі значно зростає.

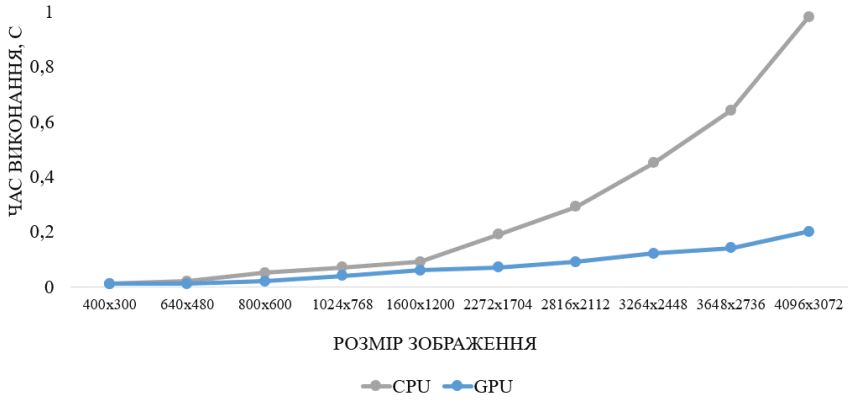


Рисунок 3 – Порівняння швидкодії роботи цифрової обробки зображення на основі CPU- та GPU-орієнтованих апаратних платформ

Таким чином, розроблено інтелектуальну систему, що призначена для підвищення якості зображень у таких сферах де недостатньо обробки

звичайними алгоритмами. Система надає можливість обробки зображень комбінаціями як із простих методів корекції (чіткості, яскравості, насиченості), так і спеціалізованих методів для певних класів зображень. Тестування відбувалось на такому класі зображень як рентгенівські знімки. В перспективі система може застосовуватись у рентгенології для обробки невдалих знімків, щоб запобігти повторному рентгенівському опроміненню людини [6,7].

ЛІТЕРАТУРА

1. Гонсалес Р. Цифровая обработка изображений / Рафаэл С. Гонсалес, Ричард Е. Вудс; пер. с англ. Л. И. Рубанова, П. А. Чочиа ; науч. ред. П. А. Чочиа. - 3-е изд., испр. и доп.. - Москва: Техносфера, 2012. - 1103 с. – ISBN 9785948363318.
2. Гашников М.В. Методы компьютерной обработки изображений: учебное пособие / Гашников М.В., Глумов Н.И., Ильясова Н.Ю. и др.; Под ред. В. А. Сойфера. – Москва: ФИЗМАТЛИТ, 2001. – 784с. – ISBN 5-9221-0180-3.
3. Поляков А. Ю. Методы и алгоритмы компьютерной графики в примерах на Visual C++ и C#: научное издание / А.Ю. Поляков, В.А. Брусенцев. – 2-е изд., перераб. и доп. – СПб.: БХВ -Петербург, 2003. – 547 с. – ISBN 5941573774.
4. Яшнин В. В. Анализ и обработка изображений: принципы и алгоритмы / Яшнин В. В. – М.: Машиностроение, 2004. – 112 с. – ISBN 5-217-02625-1.
5. Яровий А.А. Аналіз методів і моделей цифрової корекції та підвищення якості растрових зображень у сфері рентгенографії / А.А. Яровий, Д.Г. Пасічник // Вимірювальна та обчислювальна техніка в технологічних процесах. – 2016. – №3(56). – С. 136-142.
6. Яровий А.А. Проектування системи цифрової корекції та підвищення якості растрових зображень у сфері рентгенографії / Яровий А.А., Арсенюк І.Р., Пасічник Д.Г. // Інформаційні технології та комп'ютерна інженерія. – 2017. – №1(38). – С. 72-77.
7. Свідоцтво про реєстрацію авторського права на твір № 68765. Комп'ютерна програма підвищення якості растрових зображень / Яровий А.А., Пасічник Д.Г. Дата реєстрації Державною службою інтелектуальної власності України 24.11.2016.

Отримано 26.06.2017

УДК 004.92

ПРОГРАМНЕ ПІДКЛЮЧЕННЯ СЕНСОРУ KINECT ДО КОНСТРУКТОРУ GAME MAKER

А.К. Шумейко, Є.О. Башков

Донецький національний технічний університет, м. Покровськ

В роботі висвітлюється підхід до розширення можливостей керування процесом гри, створеної в конструкторі Game Maker за допомогою використання сенсору руху Microsoft Kinect.

Виконується аналіз сучасних засобів для взаємодії користувача з ігровою середою, розглядаються особливості роботи з Kinect API та способи отримання інформації за різних вихідних потоків сенсора. Наводяться відомості щодо створеної динамічної бібліотеки, особливостей реалізації та способів розробки ігрового додатку на базі системи Game Maker з використанням Microsoft Kinect.

Ключові слова: комп'ютерні ігри, сенсор Microsoft Kinect, програмна платформа Game Maker, динамічна бібліотека.

1. ВСТУП

В сучасному геймдизайні важливу роль відіграє інтерфейс між користувачем (гравцем) та ігровим додатком. Можливості сучасних ігрових інтерфейсів, їх переваги та недоліки розглянуто в [1], [2], [3]. Серед сучасних засобів взаємодії користувача і комп'ютера, що находять широке застосування в іграх, слід виділити сенсори руху. Вони мають наступні переваги:

- розпізнавання жестів та розпізнавання голосу;
- можливість роботи у темряві;
- можливість розпізнавати сидячі фігури;
- компактність, простота і доступність;
- відкритий код для розробників, за допомогою якого вони можуть створювати буквально все, що душі завгодно;
- розширені можливості управління персонажами;
- переніс гри у реальний світ, що в комбінації зі шлемами віртуальної реальності дає ефект повного занурення;
- активна участь гравця, його фізичне навантаження.

GameMaker - один з найпопулярніших ігрових движків, потужний інструментарій для професійної розробки 2D ігор. Конструктор GameMaker підтримує розробку розширень під безліч платформ на відповідних їм мовами.

Об'єднання переваг і можливостей GameMaker [4] та сенсору руху Microsoft Kinect [5] має створити деякий синергетичний ефект при розробці сучасних ігрових додатків.

2. СТВОРЕННЯ ДІНАМИЧНОЇ БІБЛІТЕКИ ДЛЯ ПІДКЛЮЧЕННЯ MICROSOFT KINNECT

Бібліотека динамічного компонування для підтримки передачі даних від сенсору до Game Maker створено в IDE Visual Studio [6].

У створеній бібліотеці додано необхідні нам бібліотеки: "NuiApi.h", "NuiImageCamera.h", "NuiSensor.h", "NuiSkeleton.h". Розглянемо функції, використовувані у бібліотеці.

Перша функція – це конструктор класу CSkeletonBasics, який ініціалюзує об'єкт класу скелета. Надалі цей клас буде використовуватись для збереження даних, отримуваних з сенсора Kinect.

Друга функція – деструктор класу CSkeletonBasics. Він використовується для звільнення пам'яті, зайнятої програмою. В основі лежить функція SafeRelease, за допомогою якої ми можливо безпечно зупинити роботу сенсора.

Данні функції є внутрішніми та не передаються до системи Game Maker, тому вони на мають модифікатора GMEXPORT.

Третя функція отримує кількість під'єднаних сенсорів. Вона повертає значення -1, якщо не підключено жодного сенсора. Інакше – повертається кількість сенсорів, розпізнаних системою.

Четверта функція – відповідає за ініціалізацію сенсора. Їй передається кількість сенсорів, та для кожного знайденого сенсора створюється його зразок. Після створення зразка перевіряється статус знайдених сенсорів функцією NuiStatus(), котра в результаті повертає структуру HRESULT.

П'ята функція – перевіряє, чи побудовано скелет гравця. Для цього перевіряється готовність даних. Для отримання координат певних точок тіла потрібно вилучити координати з точки скелета, та перевести в структуру Vector4. Для вибору точки тіла вказується її номер або повна назва.

Розроблена бібліотека дозволяє перевіряти наявність підключених сенсорів Kinect, та виводить їх кількість, ініціалізувати доступні сенсори, готовить дані скелета для обробки та отримувати координати обраних частин тіла.

На рисунку 1 наведено приклад візуалізації скелету гравця.

3. РОЗРОБКА ДОДАТКА В GAME MAKER

Для перевірки функціональних можливостей бібліотеки розроблена програма в середовищі Game Maker. Програма демонструє можливість оброблювати дані, що надходять з сенсора Kinect. Стандартний процес створення типової гри в Game Maker наведено, наприклад, в [4].

Розроблена бібліотека підключається в процесі створення об'єкту при завданні дій. Це можна зробити за допомогою функції:

external_define(dll, name, calltype, restype, argnumb,
argtype[0], argtype[1], ...argtype[10]) ;



Рисунок 1 - Керування персонажем за допомогою Microsoft Kinect

Ця функція може бути використана для визначення виклику функції зовнішньої бібліотеки DLL (для Windows) або dylib (для Mac). Першими параметрами є ім'я (і шлях) файлу, та ім'я функції. Також необхідно визначити угоду про виклики, які будуть використовуватися, а також тип результату та задати число аргументів, що функція може приймати (від 0 до 15) і для кожного з аргументів необхідно вказати його тип.

Після ініціалізації функції бібліотеки, необхідно створити подію Game Start. Це спеціальна подія спрацьовує тільки один раз за всю гру. Подія повинна бути представлена в першій кімнаті гри. Слід зазначити, що ця подія відбувається після створення події всіх примірників і тому може містити код або дії зі змінними, які були

раніше визначені. Ця подія, як правило, визначається тільки в одному «контролерному» об'єкті, виконується один раз на початку гри та використовується для ініціалізації глобальних змінних.

На початку гри також необхідно ініціалізувати сенсор Kinect та впевнитись, що є можливість отримувати дані. Для цього виконується виклик

external_call (id, args[0...15]).

В якості параметрів передаються ім'я бібліотечної функції, що викликається, та її параметри.

Перевіривши наявність сенсора та його готовність до роботи, створюється подія Draw. Ця подія завжди буде викликатися, поки екземпляр об'єкту має видимий прапор. Таким чином кожен крок гри буде викликатись подія малювання.

На рисунку 2 наведено приклад передачі координат рук гравця до ігрового додатку. На рисунку 3 показано зовнішній вигляд розробленої демонстраційної гри, що використовує можливості сенсора Kinect.

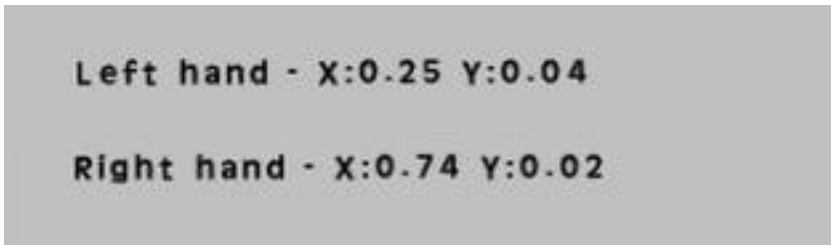


Рисунок 2 - Приклад передачі координат рук гравця до Game Maker

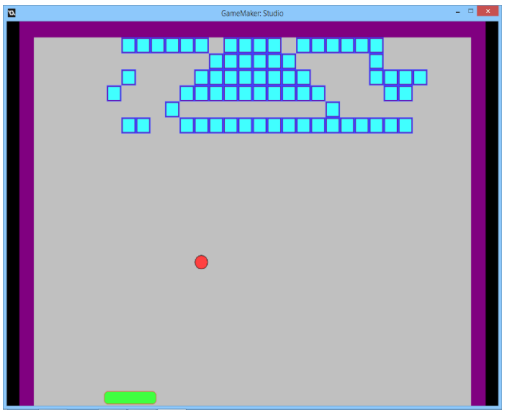


Рисунок 3 - Зовнішній вигляд розробленої демонстраційної гри

4. ВИСНОВКИ

Розроблено додаток для розширення функціональних можливостей інтерфейсу взаємодії системи Game Maker та користувача, через забезпечення обробки потоків даних з сенсора Kinect. Додаток має такі функції:

- Підключення власної динамічної бібліотеки, написаної мовою C++.
- Отримання інформації о підключених сенсорах.
- Ініціалізація сенсора для роботи з необхідними потоками даних.
- Отримання координат зчленувань тіла та передача їх ігровому додатку.

В перспективах розвитку додатку є покращення якості отримання інформації, розробка ігор з використанням навиків, отриманих в результаті виконання дипломної роботи, написання навчальних матеріалів для розробки ігор з використанням сенсора Kinect.

ЛІТЕРАТУРА:

[1] [електронний ресурс] // Режим доступу: https://en.wikipedia.org/wiki/User_interface
[2] [електронний ресурс] // Режим доступу: https://en.wikipedia.org/wiki/Text-based_user_interface
[3] [електронний ресурс] // Режим доступу: https://en.wikipedia.org/wiki/Game_controller
[4] [електронний ресурс] // Режим доступу: https://www.yoyogames.com/gamemaker
[5] [електронний ресурс] // Режим доступу: https://developer.microsoft.com/en-us/windows/kinect/develop
[6] [електронний ресурс] // Режим доступу: https://habrahabr.ru/company/microsoft/blog/323452

Отримано 29.06.2016

УДК 004.932

АНАЛИЗ МЕТОДОВ ПОИСКА НЕЧЕТКИХ ДУБЛИКАТОВ В БАЗЕ ДАННЫХ ИЗОБРАЖЕНИЙ

А.А.Ермолов, Н.С.Костюкова
Донецкий национальный технический университет, г.Покровск

Розглянуто алгоритми пошуку нечітких дублікатів ORB, BRISK, SIFT, SURF, що ґрунтуються на аналізі ключових точок, та виконано їх порівняння з точки зору часових характеристик, повноти й точності пошуку.

Ключові слова: порівняння зображень, нечіткі дублікати, ефективність пошуку.

ВВЕДЕНИЕ

Задача распознавания нечетких дубликатов изображений актуальна для систем технического зрения, поиска изображений по образцу в сети Интернет, цифровых архивах и библиотеках, при анализе космических и аэрофотоснимков и во многих других сферах.

Под нечеткими дубликатами понимаются изображения, которые могут быть переведены друг в друга путем таких преобразований, как поворот, сдвиг, изменение угла зрения, изменение масштаба, изменение освещения и других. Такие изображения могут быть получены в результате изменения условий их съемки или при редактировании.

Задача поиска нечетких дубликатов является нетривиальной - до сих пор не существует универсальных методов и алгоритмов, позволяющих безошибочно выявить нечеткие дубликаты изображений и при этом иметь небольшую вычислительную сложность.

АЛГОРИТМЫ, ПОЗВОЛЯЮЩИЕ ВЫПОЛНЯТЬ ПОИСК НЕЧЕТКИХ ДУБЛИКАТОВ

Сложность задачи поиска нечетких дубликатов обусловлена необходимостью представления изображений в формате, который обеспечивает инвариантность по отношению к потенциально сложных геометрических и фотометрических преобразований.

Для построения модели изображения могут быть использованы как глобальные, так и локальные свойства изображения.

Существующие модели, основанные на глобальных характеристиках качества изображения, в общем случае не гарантируют эффективности идентификации нечетких дубликатов. Так, использование свойств цвета, например, матрицы изменения

яркости, может приводить к ошибкам в случае изменения условий освещения сцены или использования определенных фильтров.

С другой стороны, использование характерных локальных признаков, устойчивых к геометрическим и фотометрических преобразований, представляется более приемлемой альтернативой. Одним из подходов, основанных на использовании локальных свойств изображений, есть методы обнаружения точечных особенностей.

Точечной особенностью («точкой интереса» или «ключевой точкой») называется точка изображения, структура окрестности которой ковариантная заданным преобразованием изображения.

Для выявления точечных особенностей существуют методы детекторы. Результатом работы детекторов есть множество особых точек, для которых рассчитываются дескрипторы-векторы признаков, описывающих структуру окраине точечной особенности.

Векторы признаков строятся на основе информации об интенсивности, цвет и текстуру особой точки. Но особые точки могут представляться углами, ребрами или даже контуром объекта, поэтому, как правило, вычисления выполняются для некоторой окрестности.

Набор дескрипторов составляет модель изображения, которая впоследствии используется для сопоставления изображений.

В качестве основных алгоритмов поиска нечетких дубликатов изображений можно выделить ORB, BRISK, SIFT, SURF. Данные алгоритмы были выбраны, поскольку все они базируются на сравнении ключевых точек.

Алгоритм ORB (Oriented FAST and Rotated BRIEF) опубликован в 2011 г. [1] В его основе лежит комбинация таких алгоритмов как детектор FAST (Features from Accelerated Segment Features) [2] и дескриптор BRIEF (Binary Robust Independent Elementary Features) [3] с некоторыми улучшениями. Алгоритм BRISK (Binary Robust Invariant Scalable Keypoints) выполняет поиск особых точек также с помощью детектора FAST и дескриптора BRIEF [5], но с некоторыми модификациями. Алгоритм SIFT (Scale Invariant Feature Transform) [6] является одним из самых популярных. Он основан на использовании пирамиды Гаусса, которая строится для изображения. Преимуществом данного метода является полная инвариантность относительно масштаба, смещения и поворота. SIFT дескрипторы не лишены недостатков – не все полученные точки и их дескрипторы будут отвечать заявленным требованиям и это может повлиять на дальнейшее решение задачи сопоставления изображений. Альтернативой SIFT является большее быстрый метод SURF (Speeded Up Robust Features) [7]. В данном методе для ускорения поиска особых точек используется интегральные изображения. Значения в каждой

точке интегрального изображения вычисляется как сумма значений в данной точке и значений всех точек, которые находятся выше и левее данной. С помощью интегральных изображений за константное время вычисляются так называемые прямоугольные фильтры, которые состоят из нескольких прямоугольных областей.

ЭКСПЕРИМЕНТЫ ПО СРАВНЕНИЮ АЛГОРИТМОВ ПОИСКА.

Для изучения возможностей указанных алгоритмов использована тестовая база, состоящая из 300 изображений, в которую входят изображения животных, строений, пейзажей, предметов, людей, растений. База содержит как естественные, так и искусственные дубликаты. Изображения в базе имеют различное разрешение и форму (вертикальные, горизонтальные, квадратные).

Для анализа работы выбранных алгоритмов была разработана программа в среде разработки Qt Creator средствами языка C++. В качестве эталонных реализаций алгоритмов была использована библиотека OpenCV 3.2.0.

Программа позволяет выполнить обработку множества изображений любым из указанных алгоритмов с целью поиска нечетких дубликатов. Сравнивались временные характеристики работы алгоритмов. Для оценки качества поиска рассчитывались характеристики поиска Precision (точность) – доля подходящих изображений среди всех найденных и Recall (полнота) – доля найденных зображений среди всех. соответствующих образцу поиска [8].

На рисунках 5-6 отображены экранные формы программы, на рисунках 7, 8 – графики времени обработки, точности (precision) и полноты (recall) для каждого анализируемого алгоритма.

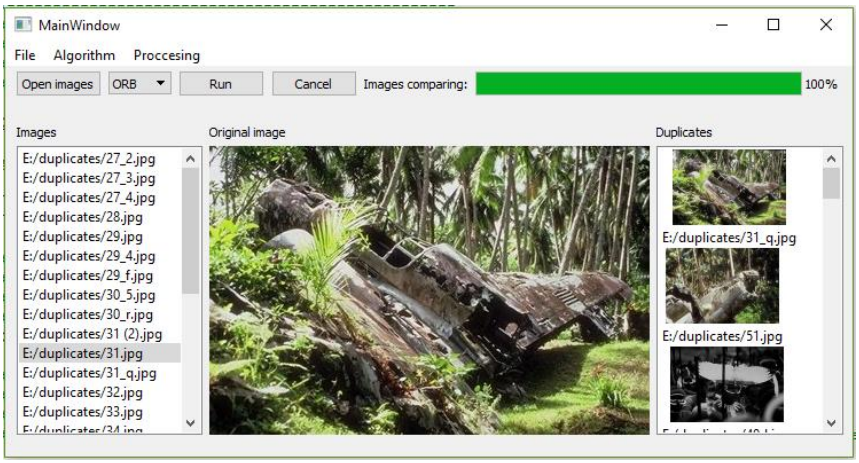


Рисунок 5 – Интерфейс программы

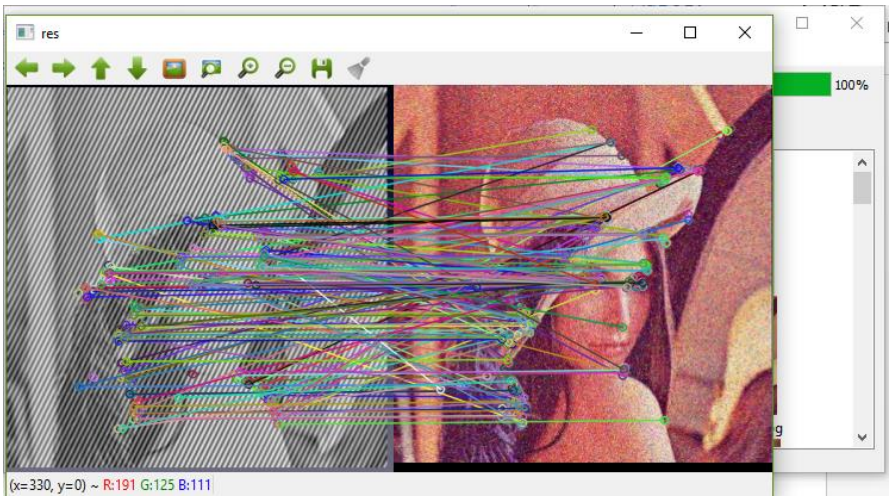


Рисунок 6 – Сравнение ключевых точек изображений

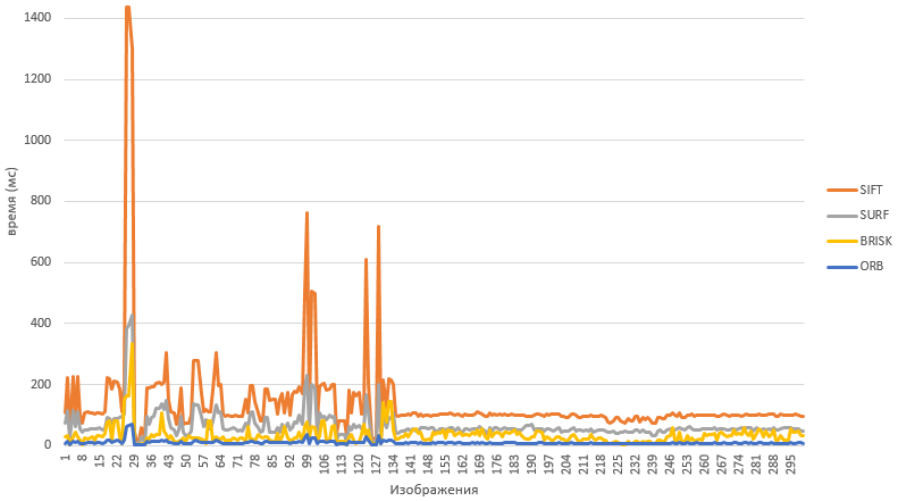


Рисунок 7 – Время работы алгоритмов

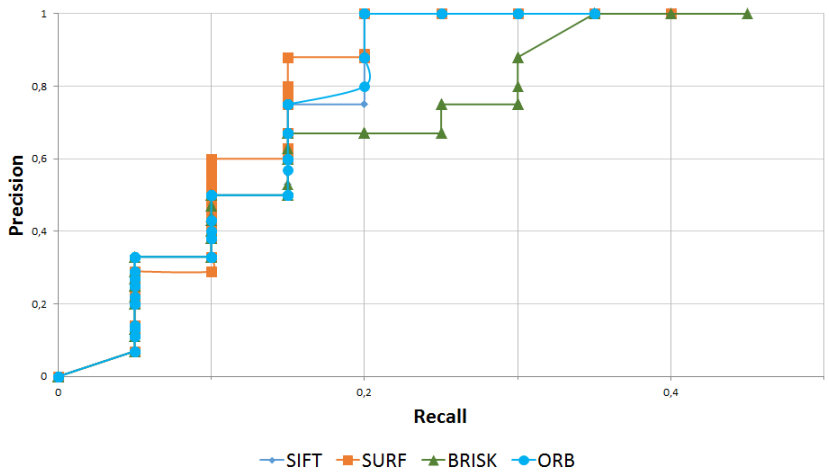


Рисунок 8 – Показатели точности (precision) и полноты (recall) поиска для разных алгоритмов

Судя по результатам анализа представленных алгоритмов, SURF обладает наилучшими показателями полноты и точности, а также лучше остальных распознает естественные дубликаты.

По показателям скорости работы наилучшим оказался ORB, а SIFT самым медленным.

Таким образом, перспективным является поиск модификаций алгоритма SURF с целью улучшения его временных характеристик.

ЛИТЕРАТУРА

1. Ethan Rublee, Vincent Rabaud, Kurt Konolige, Gary Bradski: "ORB: an efficient alternative to SIFT or SURF", Computer Vision (ICCV), IEEE International Conference on. IEEE, pp. 2564 – 2571, 2011.
2. Rosten, Edward, Tom Drummond: "Machine learning for high-speed corner detection", 9th European Conference on Computer Vision (ECCV), pp. 430 – 443, 2006.
3. Michael Calonder, Vincent Lepetit, Christoph Strecha, Pascal Fua, "BRIEF: Binary Robust Independent Elementary Features", 11th European Conference on Computer Vision (ECCV), pp. 778 – 792, 2010.
4. Harris, C., Stephens, M.: "A Combined Corner and Edge Detector". Proceedings of the 4th Alvey Vision Conference, pp. 147 – 151, 1988.
5. Stefan Leutenegger, Margarita Chli, Roland Siegwart: "BRISK: Binary Robust Invariant Scalable Keypoints". Computer Vision (ICCV), pp. 2548 – 2555, 2011.
6. Lowe, David G. (1999). "Object recognition from local scale-invariant features" (PDF). Proceedings of the International Conference on Computer Vision. 2. pp. 1150–1157
7. Speeded-Up Robust Features [Електронний ресурс]. – Режим доступу: ftp://ftp.vision.ee.ethz.ch/publications/articles/eth_biwi_00517.pdf.
8. . Konstantinidis K., Gasteratos A., Andreadis I. Image retrieval based on fuzzy color histogram processing. Optics Communications 248 (2005), pp. 375–386.

Отримано 26.06.2017

Секція 4
Моделювання систем та об'єктів промислової автоматизації

УДК 005.591.6

ВИКОРИСТАННЯ TATU SMART LAB У ПІДГОТОВЦІ ТА ПЕРЕПІДГОТОВЦІ ФАХІВЦІВ З ПРОМИСЛОВОЇ АВТОМАТИЗАЦІЇ

В.Я. Воропаєва, к.т.н., доц.
Державний вищий навчальний заклад «Донецький національний технічний університет»

Розглянуто досвід виконання проекту TEMPUS «TATU: Trainings in Automation Technologies for Ukraine» (ТАТУ: «Тренінги по технологіям автоматизації для України») в Донецькому національному технічному університеті та використання лабораторного обладнання TSL (TATU Smart Lab), отриманого в рамках цього проекту, в навчальному процесі та для підвищення кваліфікації фахівців з промислової автоматизації.

Донецький національний технічний університет з грудня 2013 р. бере участь у проекті технічної допомоги Європейського Союзу 544010-TEMPUS-1-2013-1-DE-TEMPUS-JPHES «Тренінги по технологіям автоматизації для України» (TATU: Trainings in Automation Technologies for Ukraine) [1]. Партнерами в проекті виступають ще 4 українських ВНЗ (Івано-Франківський національний технічний університет нафти і газу, Одеська національна морська академія, Одеський національний політехнічний університет, Харківський національний університет радіоелектроніки), 5 європейських університетів (Університет прикладних наук Дюссельдорф – Німеччина, Університет прикладних наук Каринтії – Австрія, Центр досліджень автоматики університету Лімерика – Ірландія, Університет Антверпен – Бельгія, Політехнічний університет Валенсії – Іспанія) та Інститут промислової інженерії (Фраунхофер) – Німеччина.

За три з половиною роки реалізації проекту українські університети-партнери провели дослідження вітчизняного ринку освітніх послуг в галузі промислової автоматизації, спільно з європейськими партнерами розробили проект лабораторного стенду TSL (TATU Smart Lab) з використанням обладнання трьох провідних європейських вендорів (SIEMENS, PHOENIX CONTACT, BERGHOF), отримали комплекти таких TSL та впровадили їх в навчальний процес,

розробивши відповідне навчально-методичне забезпечення. Між університетами-партнерами був підписаний Договір про уніфікацію надання освітніх послуг та були розподілені предметні модулі: Програмування контролерів (PC Worx і CoDeSys), Промислові мережі (Industrial Ethernet by Profinet, Industrial Ethernet by Modbus TCP, Proxy to PROFIBUS), Бездротові технології передавання даних, Системи реального часу, Стандарт OPC DA.

Ідея взаємодії полягає в тому, що кожен учасник проекту розробляє програму та навчально-методичні матеріали двох рівнів складності (базовий та експерт) для своїх модулів, обмінюється доробками з колегами і визнає результати навчання в інших центрах. Таким чином, слухач може отримати певні модулі в одному центрі, продовжити навчання в другому, а рівень експерта пройти в третьому. Або ж можливий вибір тренінгового центра за галуззю промисловості, оскільки кожний університет спеціалізується в певній сфері.

Родзинкою проекту є унікальні лабораторні стенди TSL, виготовлені за спеціальним замовленням та згідно проекту, розробленого учасниками з урахуванням сучасних тенденцій у підготовці фахівців з промислової автоматизації. Лабораторія TATU Smart Lab є мобільною, високотехнологічною та гнучкою, мультивендорною, готовою для використання в Industry 4.0. TSL складається з трьох окремих частин, кожна з яких присвячена окремій темі і може бути використана автономно або у складі лабораторного комплексу [2].

Перша частина (PLC&PROFINET Box) містить 2 програмованих логічних контролери (PLC), що можуть взаємодіяти через TCP/IP або MODBUS TCP, пристрої вводу/виводу PROFINET, керований комутатор і WLAN точку доступу. Цей блок може бути використаний при викладанні розмаїття тем – від програмування PLC до промислових комунікацій, зокрема бездротових.

Друга частина TATU Smart Lab називається PROFIBUS Box і складається з пристроїв вводу/виводу PROFIBUS, PROFINET проксі для PROFIBUS DP з вбудованим вимикачем і точкою доступу. На цьому обладнанні зручно вивчати особливості технології PROFIBUS. При цьому PROFIBUS Box може використовуватись окремо або у поєднанні з PLC&PROFINET Box. Для цього обладнання має необхідні інтерфейси та клемні модулі.

Третя частина TSL називається «Стенд моделювання процесів» і призначений для опанування технологій імітаційного моделювання технологічних процесів. Моделі мають бути розроблені у середовищі CoDeSys та завантажені на компактний контролер EC2250 EtherCAT. Обладнання дозволяє використовувати веб-інтерфейс для віддаленого

контролю за процесом моделювання та має велику кількість різноманітних входів і виходів для організації тестування моделей.

Донецький національний технічний університет розробляє навчально-методичний комплекс для предметних модулів за темою промислових мереж (Industrial Ethernet by Modbus TCP, Proxy to PROFIBUS), але, звичайно, базові знання з програмування контролерів, зокрема вміння працювати в PC Worx і CoDeSys є необхідним стартовим рівнем для опанування подальшої інформації. Отже в освітньому процесі ми використовуємо не тільки власні доробки, але й навчально-методичні матеріали наших партнерів [3, 4]. Отримане в рамках проекту обладнання та розроблене ДонНТУ і університетами-партнерами навчально-методичне забезпечення використовується в Донецькому національному технічному університеті при підготовці магістрів та бакалаврів не тільки спеціальності «Автоматизація та комп'ютерно-інтегровані технології», але й для студентів, що навчаються за спеціальностями «Телекомунікації та радіотехніка» і «Електроніка».

Під час виконання проекту командою ТАТУ ДонНТУ, до якої входять викладачі факультету Комп'ютерно-інтегрованих технологій, автоматизації, електроінженерії та радіоелектроніки, переглянуті програми трьох навчальних курсів для магістрів та 5 для бакалаврів. Для зазначених навчальних дисциплін було значно модернізовано та осучаснено перелік лекційних тем та зміст лабораторних робіт. Але, крім підвищення рівня знань фахівців-випускників, проект ТАТУ націлений на надання консалтингових та тренінгових послуг працівникам індустрії, які впровадили або планують оновити системи промислової автоматизації на виробництві. Тож на Україні була створена мережа тренінгових центрів для професійної підготовки в області технологій автоматизації.

Тож ми розробили, апробували на студентах-старшокурсниках і пропонуємо ринку курси з промислової автоматизації, реалізовані на новому сучасному лабораторному обладнанні та з використанням досвіду європейських університетів.

ЛІТЕРАТУРА:

1. Веб сайт проекта ТАТУ http://tatu.cuas.at/index.php/ua/
2. **Олег Макаров, Леонардо Рохес, Юлія Макарова.** TATU Smart Lab. Керівництво користувача. Одеса, 2017.
3. Горб С.И. и др. Программирование контроллеров в инструментальной среде: учебное пособие. – Харьков, 2017.
4. **Christian Madritsch (Editor), Wolfgang Werth (Editor).** TATU Study Book. Trainings in Automation Technologies for Ukraine. - 2017

Отримано 25.06.2017

УДК 004.032.26

СИСТЕМА ПРОГНОЗА СТАНУ ГІРНИЧОГО ОБЛАДНАННЯ НА БАЗІ МУЛЬТИАГЕНТНОГО ПІДХОДУ

Ю.Л. Дікова

Донецький національний технічний університет, Україна

В роботі розглядається централізована мультиагентна система прогнозування стану ГШО, яка складається з головного агента та сукупності підлеглих агентов, взаємодія між ними заснована на протоколі FIPA-Subscription. Основою головного агента є штучна нейронна мережа високого порядку, основою підпорядкованих агентів є штучна нейронна мережа NARMA. Запропонований підхід до побудови мультиагентних систем підвищить точність прогнозу характеристик ГШО на 9%, а ймовірність прогнозування стану - на 8%.

Ключові слова: мультиагентна система, штучна нейронна мережа, NARMA, прогноз стану.

1. ВСТУП

В процесі експлуатації на гірничо-шахтне обладнання (ГШО) впливає ряд факторів, що призводять до несправностей та відмов. Існуючі на даний час системи [1] технічного обслуговування хоч і забезпечують підтримку ГШО в справному стані, але є недостатньо досконалими. В наслідок цього питання підвищення якості діагностики та прогнозу стану ГШО стає все більш актуальним, адже старіння обладнання значно випереджає темпи технічного переоснащення.

На сьогоднішній день в якості інструменту для прогнозу широко застосовуються такі методи: регресивні і авторегресійні методи прогнозування; методи, засновані на експоненційному згладжуванні; методи на базі ланцюгів Маркова; на базі класифікаційно-регресійних дерев; штучні нейронні мережі (ШНМ). В роботі [2] наведено порівняльну характеристику перерахованих способів. Оскільки використання ІНС [3-5] при прогнозуванні дає відчутну перевагу, в роботі буде використано нейросетевой спосіб прогнозу.

Побудова способу прогнозу та діагностики на основі мультиагентної системи [6-9] дозволить забезпечити паралелізм

обробки інформації, що підвищить швидкість прогнозу стану ГШО за рахунок розпаралелювання процесів прогнозу значень його показників.

2. РОЗРОБКА СТРУКТУРИ СИСТЕМИ

В роботі пропонується централізована мультиагентна система прогнозу стану ГШО, що складається з головного агента A і набору підлеглих незалежних один від одного агентів A_i. Взаємодія між головним і підлеглими агентами здійснюється на основі протоколу FIPA-Subscribe. Кожен підлеглий агент A_i прогнозує значення температура нагріву поверхні або віброшвидкість по одній з трьох осей.

БД головного агента A зберігає вектор останніх спрогнозованих значень стану ГШО, а БЗ зберігає набір асоціативних правил ШНМ.

БД кожного підлеглого агента A_i зберігає вектор останніх виміряних значень i-ї ознаки та відповідний йому вектор останніх спрогнозованих значень i-ї ознаки. БЗ кожного підлеглого агента A_i зберігає набір асоціативних правил для i-ї ШНМ.

Алгоритм роботи головного агента полягає в наступному: агент A запускає всіх підлеглих агентів A_i, потім посилає їм запит про підписку на отримання спрогнозованих значень ознак та чекає на відповідь-підтвердження від кожного підлеглого агента. Далі головний агент повідомляє оператору результат відповідей. У випадку успішного підтвердження виконується збір прогнозованих значень $y_i(n+1)$ від підлеглих агентів та відбувається прогноз стану ГШО $z(n+1)=F(y_1(n+1),...,y_{21}(n+1))$, використовуючи запропоновану автором ШНМ. Кінцевий результат прогноза $z(n+1)$ агент A відправляє оператору.

Алгоритм роботи кожного підлеглого агента A_i полягає в наступному: агент A_i очікує, доки не отримає запит на підписку від головного агента та посилає у відповідь згоду на підписку. Далі агент A_i з квантом часу Δt отримує значення показника $u_i(n)$ від відповідного датчика. Якщо модуль різниці між заміряним значенням ознаки $u_i(n)$ та спрогнозованим раніше значенням ознаки $y_i(n)$ перевищує заданий поріг, то агент A_i повідомляє головному агенту A про збій в підписці. У випадку коректної роботи агент A_i виконує прогноз значення ознаки

$y_i(n+1) = F_i(u_i(n),...,u_i(n-M), y_i(n),...,y_i(n-M))$, використовуючи i-у ШНМ. Спрогнозоване значеннння ознаки $y_i(n+1)$ посилається головному агенту.

Оскільки в основі роботи головного агента лежить комплексний прогноз стану ГШО, було прийнято рішення використовувати авторську ШНМ високих порядків, структурна схема якої приведена на рис 1.

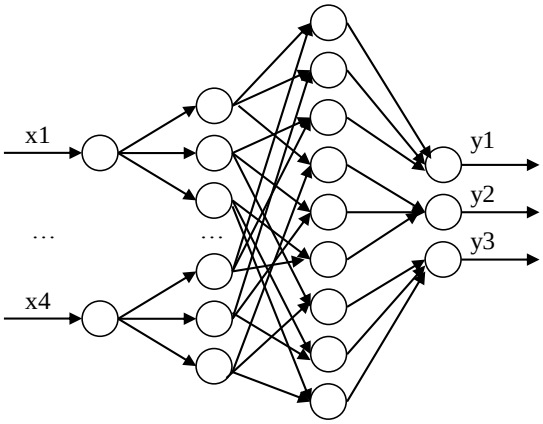


Рисунок 1 – ШНМ високих порядків головного агента

Модель запропонованої ШНМ представлена у вигляді:

$$y_j^{(1)} = \begin{cases} \dfrac{1}{1+\exp(b_{1j} - a_{1ij}x_i)} & j = (i-1)N^{(3)} + 1 \\ \dfrac{1}{1+\exp(b_{2j} - a_{2ij}x_i)} + \dfrac{1}{1+\exp(d_{2j} - c_{2ij}x_i)} & j = (i-1)N^{(3)} + 2, \\ \dfrac{1}{1+\exp(b_{3j} - a_{3ij}x_i)} & j = (i-1)N^{(3)} + 3 \end{cases}$$

де $j \in 1, N^{(1)}$, $i \in 1, N^{(0)}$.

$$y_j^{(2)} = f^{(2)}(s_j^{(2)}) = s_j^{(2)}, \; j \in 1, N^{(2)},$$

$$s_j^{(2)} = b_j^{(2)} + \sum_{i_1=1}^{N^{(1)}} ... \sum_{i_{N^{(0)}}=1}^{N^{(1)}} w_{i_1...i_{N^{(0)}}j}^{(2)} y_{i_1}^{(1)} * ... * y_{i_{N^{(0)}}}^{(1)},$$

$$y_j^{(3)} = f^{(3)}(s_j^{(3)}) = s_j^{(3)}, j \in 1, N^{(3)},$$

$$s_j^{(3)} = b_j^{(3)} + \sum_{i_1=1}^{N^{(2)}} w_{i_1 j}^{(3)} y_{i_1}^{(2)} + \sum_{i_1=1}^{N^{(2)}} \dots \sum_{i_{k=1}}^{N^{(2)}} w_{i_1 \dots i_k j}^{(3)} y_{i_1}^{(2)} * \dots * y_{i_k}^{(2)} + \dots$$

$$\dots + \sum_{i_1=1}^{N^{(2)}} \dots \sum_{i_{N^{(2)}=1}}^{N^{(2)}} w_{i_1 \dots i_{N^{(2)}} j}^{(3)} y_{i_1}^{(2)} * \dots * y_{i_{N^{(2)}}}^{(2)}.$$

де $w_{ij}^{(m)}$ - вагові коефіцієнти в шарі m, m = 1, 2, 3;

c_{pij}, a_{pij} - вагові коефіцієнти ШНМ для визначення значення функції активації $y_j^{(1)}$ в першому шарі;

d_{2j}, d_{2j} - пороги смещения ШНМ для визначення значення функції активації $y_j^{(1)}$ в першому шарі.

Навчання ШНМ проводилося на базі алгоритму зворотнього розповсюдження помилки з вчителем.

Для оцінки якості прогнозу ШНМ високих порядків було обрано критерій адекватності моделі, який означає вибір таких значень параметрів $w_i^{(2)}$, $w_i^{(3)}$, a_{ij}, b_{ij}, c_{ij}, d_{ij}, що дають мінімум середньоквадратичної помилки (різниці виходу за моделлю і бажаного виходу):

$$F = \frac{1}{P} \sum_{p=1}^{P} (y_p - d_p)^2 \to \min_{w_i^{(2)}, w_i^{(3)}, b_{ij}, a_{ij}, c_{ij}, d_{ij}}, \qquad (1)$$

де P – кількість тестових реалізацій; y_p – прогноз, отриманий по моделі; d_p – очікуваний прогноз.

Для роботи підлеглих агентів було обрано ШНМ NARMA [3], оскільки вона дає більш точний прогноз у порівнянні з іншими ШНМ [4]. Модель даної ШНМ описується наступним чином:

$$y_j^{(1)}(n) = f^{(1)}(s_j^{(1)}(n)), \; j \in \overline{1, N^{(1)}},$$

$$s_j^{(1)}(n) = b^{(1)} + \sum_{l=1}^{M^{(0)}} w_{lj}^{(1)} y^{(0)}(n-l) +$$

$$+ \sum_{l=M^{(0)}+1}^{M^{(0)}+M^{(2)}} w_{lj}^{(1)} (y^{(0)}(n-(l-M^{(0)})) - y^{(2)}(n-(l-M^{(0)}))).$$

$$y^{(2)}(n) = f^{(2)}(b^{(2)} + \sum_{i=1}^{N^{(1)}} w_i^{(2)} y_i^{(1)}(n)).$$

Навчання моделі виконувалося з використанням метаевристик [5]. Оцінка якості прогнозу ШНМ NARMA проводилася аналогічно критерію (1).

Результати прогнозу запропонованих мереж детально розглядалися та порівнювалися в роботі [10].

3. ВИСНОВКИ

Запропонована мультиагентна система прогнозу має наступні переваги:

- спрощено поділ завдань між підлеглими агентами;
- підлеглі агенти не заважають один одному під час вирішення своїх завдань;
- підлеглим агентам не потрібно явно спілкуватися між собою під час вирішення своїх завдань;
- підлеглі агенти не дублюють дії один одного.

На основі чисельного дослідження структури ШНМ NARMA, використовуваної для прогнозу значень ознак підлеглими агентами, були обрані функції активації, визначено кількість нейронів в прихованому шарі і величина затримки, що дозволило прискорити процедуру навчання.

Запропонована ШНМ високих порядків, яка використовується для прогнозу стану головним агентом, володіє наступними перевага:

- можливість використання в одному шарі різних функцій активації, що дозволяє врахувати особливості ознак (характеристик) стану ГШО;
- відсутність необхідності визначення кількості прихованих шарів і кількості нейронів в цих шарах, що спрощує адаптацію структури ШНМ;
- універсальність і розширюваність для будь-якого необхідної кількості ознак (характеристик) стану ГШО.

Адаптація параметрів запропонованих ШНМ виконувалася за допомогою комбінації локального і випадкового пошуку, ймовірність виконання яких залежала від номера ітерації. Це дозволило збільшити точність прогнозу значень показників ГШО на 9%, а ймовірність прогнозу стану ГШО на 8%.

Запропонований підхід до побудови мультиагентних систем може бути використаний для комп'ютерних систем прогнозу загального і спеціального призначення.

ЛІТЕРАТУРА

[1] **Вовна А.В.** Компьютеризированная информационно-измерительная система контроля концентрации метана в угольных шахтах: дис. … кандидата технических наук: 05.13.05 – компьютерные системы и компоненты / Вовна Александр Владимирович. – Донецк, 2009. – 200 с.
[2] **Чучуева И.А.** Модель прогнозирования временных рядов по выборке максимального подобия: дис. . … кандидата технических наук: 05.13.18 – математическое моделирование, численные методы и комплексы программ / Чучуева Ирина Александровна. – Москва, 2012. – 153 с.
[3] **Haykin S.** Neural networks / S. Haykin. – NY: Pearson Education, 1999. – p.823.
[4] **Осовский С.** Нейронные сети для обработки информации / С. Осовский. – Пер. с польского И.Д. РудШНМкого. - М.: Финансы и статистикаю – 2002. – 344 с.
[5] **Скобцов Ю.А**. Метаэвристики: монография / Ю.А. Скобцов, Е.Е. Федоров. – Донецк: Ноулидж, 2013. – 426 с.
[6] **Городецкий В.И.** Многоагентные системы (обзор) / В.И. Городецкий, М.С. Грушинский, А.В. Хабалов // Новости искусственного интеллекта. – 1997. – № 1. – С. 12-47.
[7] **Bellifemine F.** Developing multi-agent systems with JADE / F. Bellifemine, G. Caire, D. Greenwood. – Chichester: John Wiley & Sons Ltd, 2007. –286 p.
[8] **Wooldridge M**. An Introduction to MultiAgent Systems / M. Wooldridge. – Chichester: John Wiley & Sons, Inc., 2002. – 348 p.
[9] **Федоров Е.Е.** Методология создания мультиагентной системы речевого управления: монография / Е.Е. Федоров. – Донецк: изд-во «Ноулидж», 2011. – 356 с.
[10] **Федоров Е.Е., Дикова Ю.Л**. Мультиагентная система прогноза состояния рудничной атмосферы. Труды международной конференции "Моделирование 2016". Киев, 25-27 мая 2016, с. 61-64

Отримано 20.06.2017

УДК 621.313.333

МОДЕЛЮВАННЯ ВОДОВІДЛИВНОЇ УСТАНОВКИ ЗАСОБАМИ МОБІЛЬНИХ ЛАБОРАТОРІЙ TATU

Єшан Р.В., Поцепаєв В.В., Воропаєва В.Я.
Донецький національний технічний університет м. Покровськ

Розглянуто імітаційну модель водовідливної установки, що реалізована з використанням промислового логічного контролера. Розглядається підключення моделі об'єкта до моделі системи управління та можливості використання мобільних лабораторій TATU при навчанні розробки прикладного програмного забезпечення для промислових контролерів.

Кафедра АТ ДонНТУ приймає участь в освітньому проекті TATU Tempus. В межах цього проекту ДонНТУ спеціалізується у шахтних технологіях, тому розробляються приклади об'єктів управління технологічного обладнання вугільних шахт. Такі приклади дозволять фахівцям глибше зрозуміти особливості функціонування сучасних технологій автоматизації та краще опанувати їх, нададуть широкі можливості для наочної демонстрації студентам принципів автоматизованого управління складними технологічними об'єктами та практичне опанування навиками побудови складних систем дистанційного керування.

Одним з прикладів розроблених моделей об'єктів є модель водовідливної установки. Процес відкачування води з шахтних горизонтів необхідний для забезпечення вугледобутку. Отже, важливою складовою технологічного обладнання вугільних шахт є насосні установки відкачування води. В ході виконання даного Прикладу слухачі отримують задачу розробити систему автоматизації для шахтного блоку. Метою тематичного дослідження є отримання навичок розробки і впровадження шахтних систем промислової автоматизації на базі TSL. Для вирішення поставленої задачі студенти повинні:

1. Проаналізувати особливості процесу водовідливу;
2. Сформулювати вимоги до системи управління;
3. Проаналізувати датчики, технічні засоби контролю;
4. Розробити і реалізувати проект системи управління моделі об'єкту

5. Дослідити мережевий трафік при роботі моделі

На шахті відкачування води з горизонту на горизонт і далі на поверхню шахти здійснюється багатоступеневим водовідливом. Насосна установка водовідливу кожного горизонту може розглядатися як звичайна одноступенева установка, що використовує: погружні заливальні насоси; електродні датчики рівня води у водозбірнику; датчики температури підшипників; реле подачі та тиску, для контролю заливання водою насосу перед пуском; приводи засувки; пускачі приводів засувки та заливального насосу; блоки живлення. Технологічна схема автоматизованої насосної установки водовідливу наведена на рис. 1.

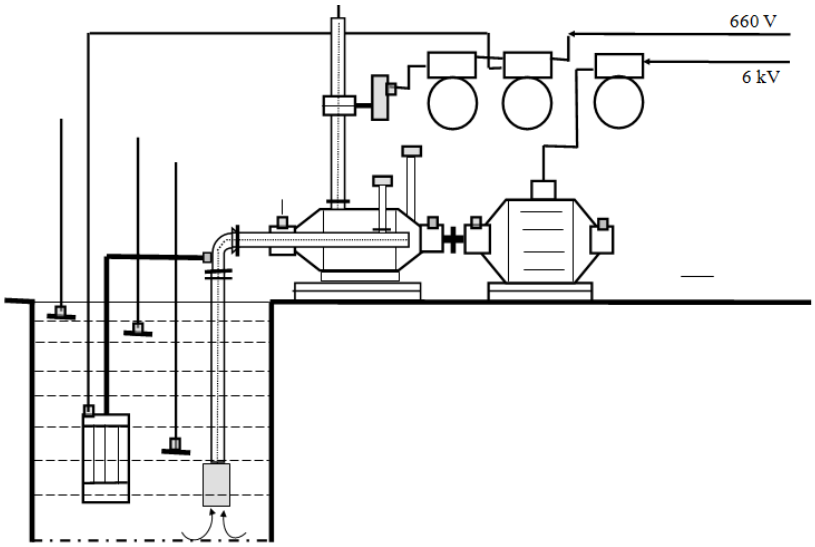


Рисунок 1 – Технологічна схема автоматизованої насосної установки водовідливу

Запуск насосної установки починається з підготовки насосних агрегатів до роботи: здійснюється заливка водою всмоктувального трубопроводу і насоса. Закінчення процесу заливки насосу необхідно контролювати. Після заливки насосний агрегат вмикається на закриту засувку на нагнітальному трубопроводі, яка відкривається тільки після набору насосом номінальних обертів. Нормальний пусковий режим забезпечує номінальний тиск у нагнітальному трубопроводі і задану продуктивність насоса. У робочому режимі потрібен контроль за процесом відкачування води, за допомогою контролювання різноманітних параметрів роботи водовідливної установки: гідравлічних та електричних. У тому числі: рівень води у

водозбірнику; замулення попереднього відстійника; тиск та подача насосу; активна потужність, споживана приводним електродвигуном. У випадку, коли рівень води підвищується і досягає аварійного рівня, означає, що приплив води до водозбірника перевищив нормальний рівень і подачі одного насосу недостатньо. Тоді, необхідно включити паралельно у роботу наступний насосний агрегат, повторивши перед цим усі операції режиму пуску, які були виконані для першого насосу. У робочому режимі можливе виникнення аварійної ситуації у випадку відхилень від нормального режиму роботи водовідливної установки. До аварійної ситуації можуть призвести ряд обставин: перегрів підшипників насосу чи приводного електродвигуна, зниження продуктивності насосу, зниження тиску у трубопроводі, досягнення водою аварійного рівня у водозбірнику. З метою уникнення негативних явищ і повернення технологічного процесу водовідливу до робочого режиму, основний насосний агрегат повинен відключатися, а до роботи має вступити резервний. Аварійна ситуація може виникнути і у процесі пуску насосного агрегату. Якщо після включення приводного електродвигуна у мережу, насос не розвиває номінальний тиск чи номінальну подачу, то засувка зачиняється і електродвигун відключається від мережі, після чого здійснюється запуск другого насосного агрегату.

Якщо процес відкачування йде нормально, то рівень води знижується до контрольованого нижнього рівня. При його досягненні треба зачинити засувку та відключити насосний агрегат.

Реалізація моделі шахтного водовідливу і системи управління ним, базується на використанні лабораторних модулів TSL ToolBox. Імітаційна модель об'єкта управління реалізована в CoDeSys програмно мовою FBD та завантажена в спеціалізований контролер Berghof EC Compact 2250. В моделі вхідні змінні визначають стан насоса (вкл / викл), а модель генерує вихідні сигнали від датчиків рівня води. Інтенсивність припливу води може бути встановлена користувачем в інтерактивному режимі (див. рис .2).

Виконана візуалізація об'єкту управління у вигляді схематичного зображення рівня води в водозбірнику, індикації стану насосів та значення датчиків рівня в CoDeSys. Для візуалізації рівня води в водозбірнику використовується інтегратор, на вхід якого подається

різниця значень припливу води і витрати від роботи насоса. Вихідний сигнал інтегратора керує розміром елемента візуалізації, що відображає рівень води. Стан насоса «включено» показується зеленим трикутником, «вимкнено» - сірим, як на рис. 3.

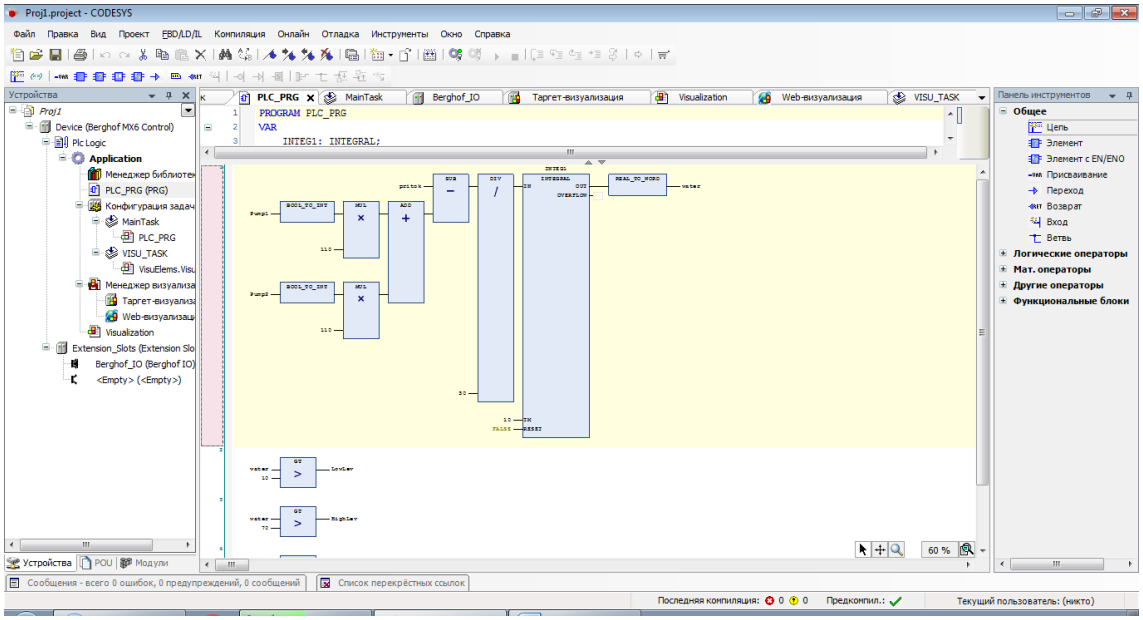


Рисунок 2 - Моделювання об'єкта з використанням CoDeSys

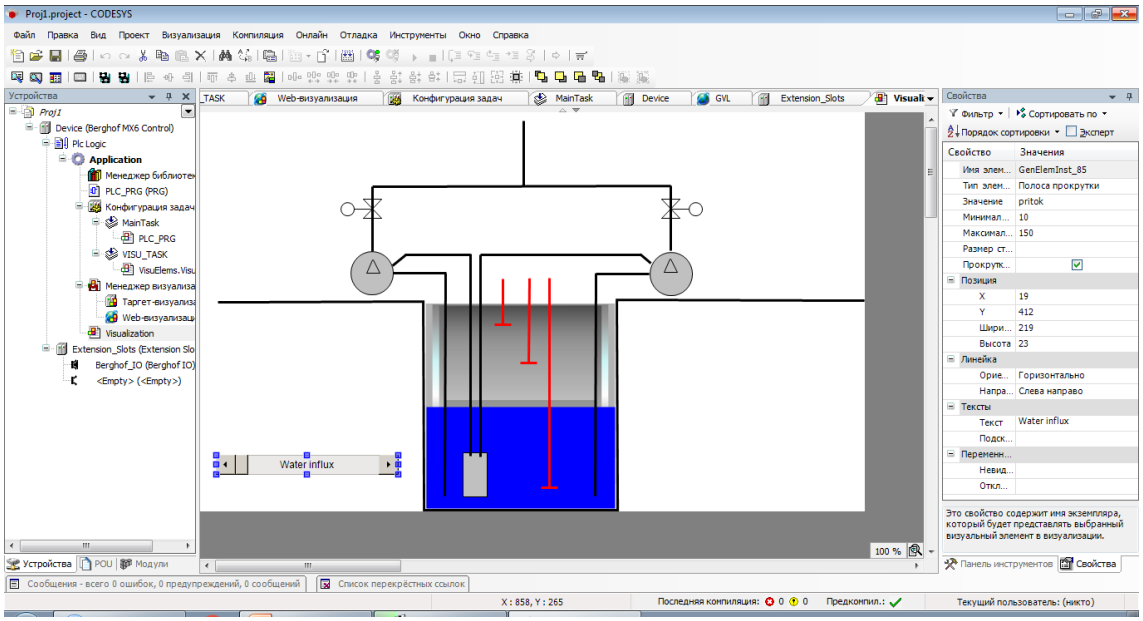


Рисунок 3 – Модель візуалізації

Для відпрацювання основних способів побудування та налаштування систем автоматичного управління промислових об'єктів, побудовано модель шахтної водовідливної установки на базі стендів TSL з контролерами та датчиками PhoenixContact. Структуру показано на рисунку 4.

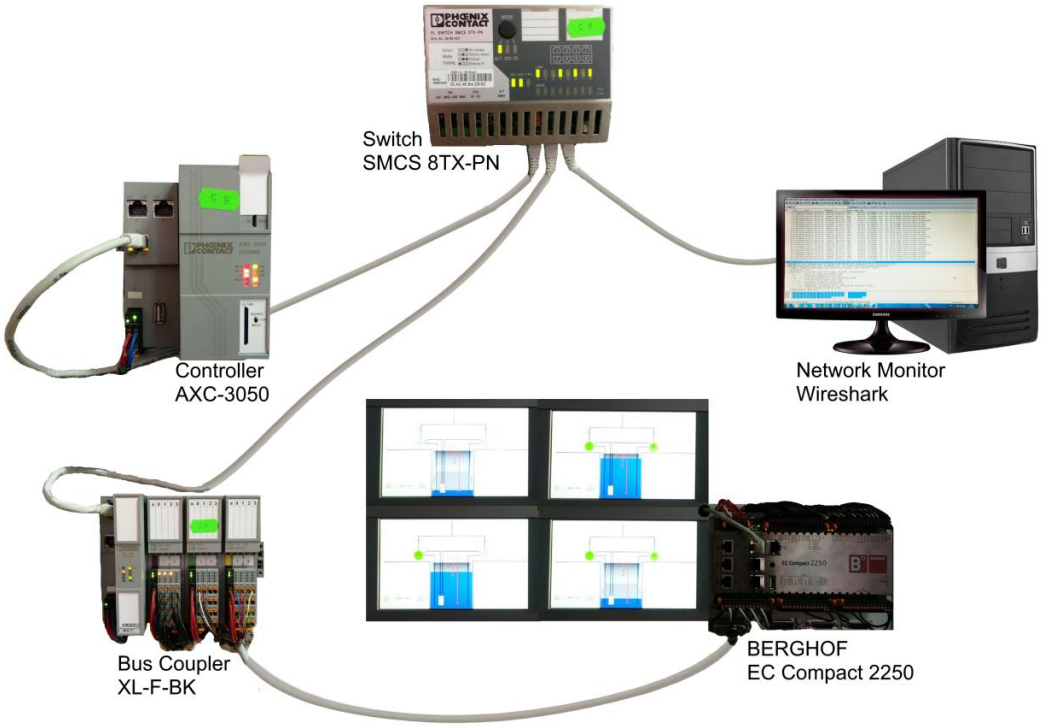


Рисунок 4 – Структура зв'язку системи управління водовідливною установкою

Студентам та слухачам курсів підвищення кваліфікації пропонується за допомогою програмного забезпечення розробника PC Worx створити спрощену систему управління насосним обладнанням та завантажити її в контролер AXC 3050.

Використання мобільних лабораторій TATU дозволяє персоналу, що практикується в розробці прикладного програмного забезпечення для промислових контролерів, наочно побачити взаємодію розробленої системи управління з об'єктом управління, та протестувати коректність роботи розроблених програм у різних режимах при різних комбінаціях вхідних сигналів. При цьому надбання досвіду не потребує підключення реального об'єкта.

УДК 621.313.333

МАТЕМАТИЧНА МОДЕЛЬ НАВАНТАЖЕНЬ НА ВИКОНАВЧИХ ОРГАНАХ ВИДОБУВНОГО КОМБАЙНА

Поцепаєв В.В., каф. «Автоматика та телекомунікації»
Донецький національний технічний університет, м. Покровськ, Україна

Розглянуто принципи побудови математичної моделі процесу руйнування вугільного пласта та формування навантажень на шнекових виконавчих органах видобувного комбайну. Програмна реалізація моделі використовується для розробки та дослідження систем автоматичного управління вугільними комбайнами.

Основною системою управління робочим режимом комбайна є система стабілізації навантаження на виконавчих органах оскільки саме вона забезпечує максимальну продуктивність, надійність та безпеку експлуатації машини.

Структурний та параметричний синтез системи стабілізації навантаження на шнеках та модернізація існуючих систем на основі сучасних контролерів робить необхідним наявність цифрової математичної моделі комбайну як об'єкта управління, зокрема моделі процесу руйнування пласта та формування моменту навантаження на шнеках. Така модель дозволить протестувати цифровий регулятор в реальному часі в максимально наближеній до реальних умов замкненій системі привід подачі – комбайн – вибій.

З огляду стану питання найбільш повною , універсальною та відповідною поставленому завданню є модель формування навантажень, що вперше розглянута в [1]. Саме вона обрана для реалізації, тому що повністю відтворює процес стужкоутворення і, отже, формування моменту навантаження на шнеках.

Принципи побудови моделі та її програмна реалізація.

Момент сил опору на виконавчому органі M і сили реакції вибою R_a, R_b , R_c в будь-який момент часу визначаються сумою моментів і сил, що діють на кожний і-тий різець, що перебуває в зоні різання:

$$M = R\sum_{i=1}^{N} z_i;\ R_a = \sum_{i=1}^{N} a_i;\quad R_b = \sum_{i=1}^{N} b_i;\quad R_c = \sum_{i=1}^{N} c_i;$$

$$a_i = -y_i \cos\gamma_i + z_i \sin\gamma_i;\quad b_i = -y_i \sin\gamma_i - z_i \cos\gamma_i;\quad c_i = x_i;$$

Тут z_i, y_i, x_i відповідно сила різання, сила подачі й бічна сила, діюча на і -тий різець, γ_i - кутове положення і -того різця; $\gamma_i = \gamma_{0i} + \gamma$,

де γ- координата виконавчого органа, одержувана з розв'язку рівнянь динаміки приводу виконавчих органів, γ_{0i}- початковий кут установки і -того різця на виконавчому органі. Значення z_i, y_i, x_i визначаються залежно від товщини стружки на кожному і -тому різці.

Товщина стружки і - того різця h_i при його заданім кутовім переміщенні γ_i визначається за виразом

$$h_i = R - \sqrt{R^2 - ((\Delta x)^2 + (\Delta y)^2)\sin^2\alpha_i} + (sign(\Delta y))\sqrt{(\Delta x)^2 + (\Delta y)^2}\cos\alpha_i$$

, де $\alpha_i = \gamma_i - arctg\frac{\Delta x}{\Delta y}$.

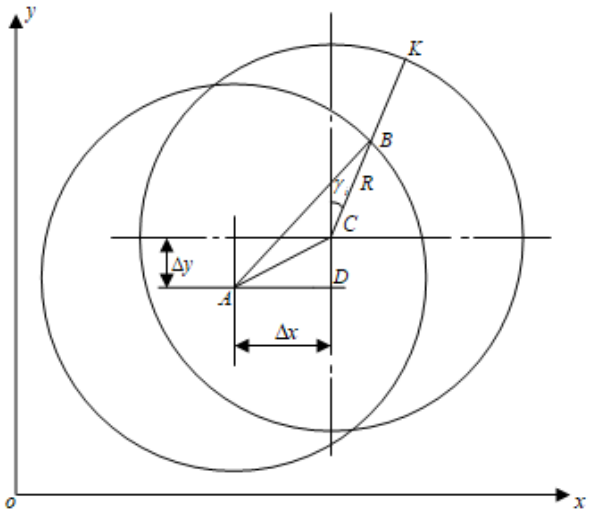


Рисунок – 1 До визначення товщини стружки на різці виконавчого органу

У формулі значення прирощень Δx, Δy, γ_i є функціями просторових координат виконавчого органу x, y, γ, отриманих з розв'язку рівнянь руху комбайна в просторі лави й рівнянь динаміки приводу виконавчих органів:

$$\Delta x = x - x_n,\ \Delta y = y - y_n,\ \gamma_i = \gamma + \gamma_{0i},$$

де x, y і x_n, y_n - відповідно абсциса й ордината виконавчого органу в даний момент часу t й у момент часу t_n, коли різець, що йде спереду та перебуває в одній лінії різання з розглянутим і - тим різцем, перетинав радіус, що має кут нахилу γ_i; γ - кутова координата виконавчого органа в момент часу t.

Для одного різця в лінії різання, різець, що йде спереду, буде сам же розглянутий різець, і для знаходження збільшень Δx, Δy необхідно знати його ж координати x_n й y_n на попередньому оберті виконавчого органа. Отже, при програмній реалізації формули на кожному кроці обчислення h_i у загальному випадку необхідно мати передісторію руху комбайна x_n, y_n за один оберт виконавчого органу.

У пропонованому алгоритмі формування передісторії і обчислення h_i виконується через кут $\Delta\varphi$, що знаходиться як $\Delta\varphi = 2\pi / p$, де p - число положень за один оберт виконавчого органу, у яких обчислюються значення h_i. Значення p обумовлене збереженням частотних властивостей і форми спектра навантажень на виконавчому органі. По значенням h_i обчислюються зусилля на різцях, що контактують з вибоєм, і далі на виконавчому органі. Розрахунок зусиль в кожному положенні шнеку закінчується занесенням поточних координат x і y в рядок масивів передісторії для подальшого використання на наступному оберті виконавчого органу.

На теперішній стадії досліджень розглянутий алгоритм реалізовано програмно у вигляді m-функції. Програмна реалізація розглянутого методу виконана для комбайна із двома виконавчими органами, що дозволяє моделювати будь-яку кінематичну схему розташування шнеків.

На рис. 2 наведено моделювання реакції моментів навантаження на випереджаючому шнеку – М1, відстаючому – М2 та сумарного моменту на шнеках М на стрибки швидкості подачі – v від 3 м/хв до 2 м/хв на 2 секунді та до 4 м/хв на 4 секунді для потужності пласта 0,8 м та опірності пласта різанню 250 Н/мм.

На рис. 2 наведено дослідження роботи комбайну під управлінням системи стабілізації моменту навантаження на виконавчих органах. В модельному експерименті досліджується реакція системи на стрибки опірності різанню, що відпрацьовуються зміною швидкості подачі комбайна. В показаному фрагменті початкова опірність дорівнює 350 Н/мм, на 15-тій секунді опірність знижено до 250 Н/мм, на 25-тій секунді - зниження до 200 Н/мм, на 35-тій секунді - підвищення до 250 Н/мм, на 45-тій секунді - підвищення до початкової опірності 350 Н/мм.

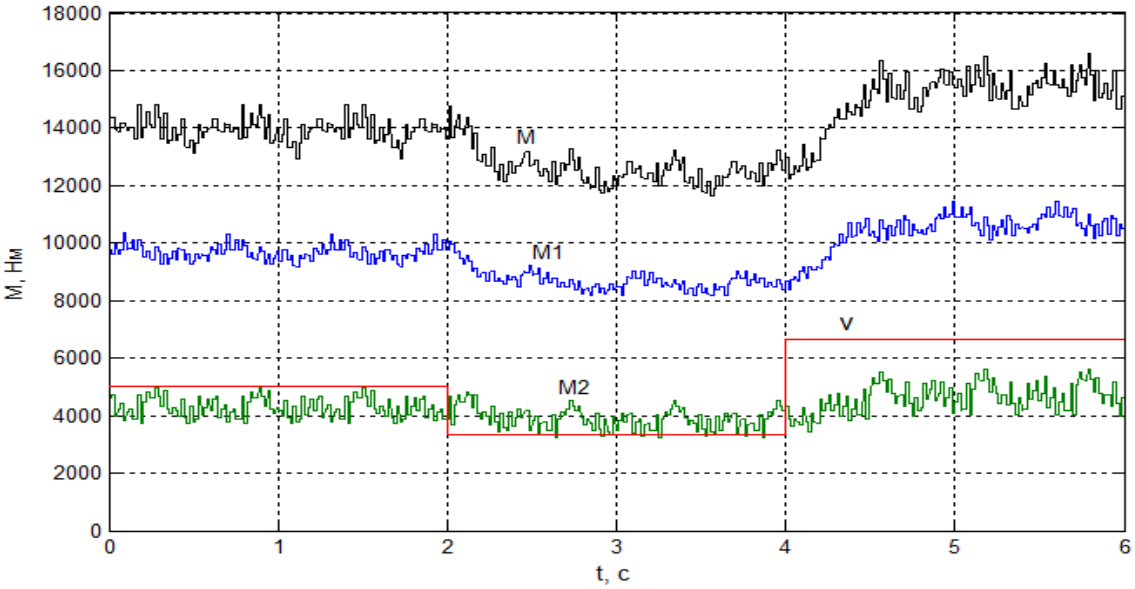


Рисунок – 2. Моделювання реакцій навантажень на шнеках

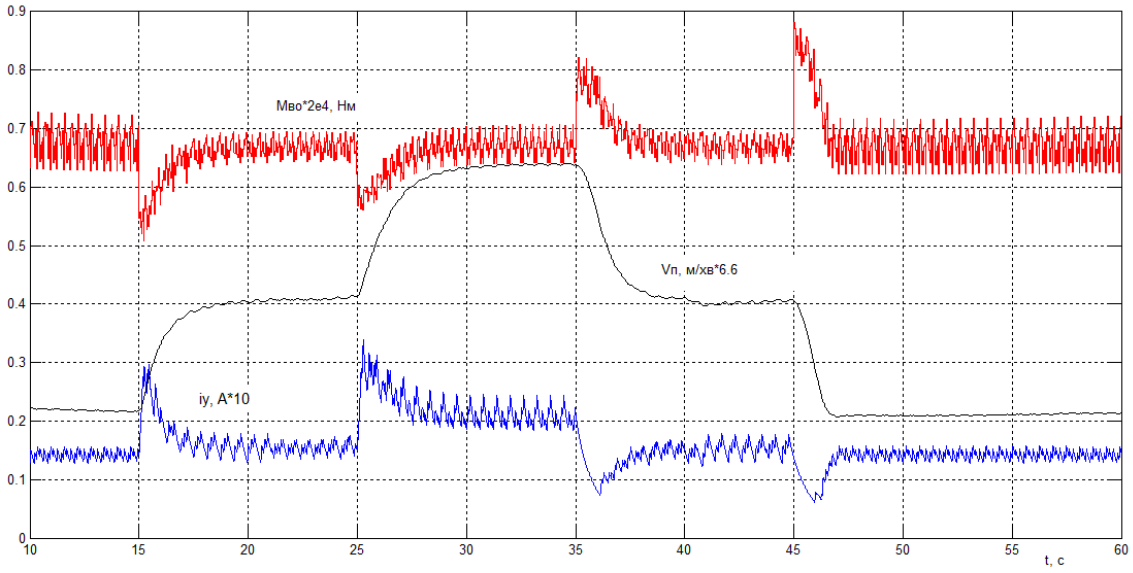


Рисунок – 3. Відпрацювання системою стабілізації навантаження на шнеках збурення по опірності пласта різанню

ЛІТЕРАТУРА

1. Усовершенствованный метод имитации динамических нагрузок привода резания очистного комбайна / В.Е. Бурдин, Б.Д. Либерман, В.В. Поцепаев, В.Л. Советов. - Науч. тр. / ИГД им. А.А. Скочинского, 1984, вып. 228. Параметры и надежность машин для подземной добычи угля, с. 57 – 65.

Отримано 29.06.2017.

УДК 004.457

СУЧАСНІ ЗАСОБИ НАВІГАЦІЇ У ВИРІШЕННІ КОМЕРЦІЙНИХ ЗАВДАНЬ, ПРОГРАМА "ROUTE"

Тихонова О.А., Скрипник Т.В.

Донецький національний технічний університет (ДонНТУ)

У даній доповіді наводиться вирішення проблеми щодо обліку і відстеження комерційного транспорту сучасними засобами навігації. Авторська програма Route вирішує проблему надлишковості інформації, накопиченої за кілька днів.

Ключові слова: GPS, GPS-регістратор, електронна карта, програма OZIEXPLORER, маршрут, файл, програма Wintec G_Tool, координати, Delphi 7

1. ВСТУП

В даний час, все частіше виникає необхідність фіксувати маршрути пересування будь-яких об'єктів або визначати їх знаходження в потрібний момент часу. З одного боку велика кількість людей отримують задоволення в подорожах, з іншого боку, будівельні компанії, компанії з перевезення вантажів, людей і т.д. хотіли б фіксувати переміщення власних рухомих об'єктів.

Сьогодні подібні задачі вирішуються використанням електронних засобів навігації, таких як GPS-приймачі.

GPS – глобальна система позиціонування (Global Positioning System). Ця система заснована на наявності сузір'я супутників, кожен з яких в будь-який момент часу має чітко визначені координати. В даний час існує дві системи: американська NAVSTAR і російська ГЛОНАСС. [1] Російська система ще недостатньо доступна для цивільних споживачів, тому основним ресурсом є американська система NAVSTAR, і саме її для стислості називають терміном GPS.

GPS- приймач – радіоприймальний пристрій для визначення географічних координат поточного місцезнаходження на основі даних про тимчасові затримки приходу радіосигналів, що випромінюються супутниками групи NAVSTAR. На базі GPS-приймачів створюються як самостійні пристрої - GPS-навігатори, GPS-трекери, що мають власний процесор для необхідних розрахунків і (в основному, у навігаторів) дисплей для відображення інформації. Так і GPS-

приставки (або GPS-реєстратори) до КПК і ноутбуків, які бувають бездротові (BlueTooth) і дротові (USB, RS-232) [2].

Електронні карти традиційно діляться на два основних типи: растрові і векторні.

Растрові карти представляють собою звичайне зображення, подібне до фотографії з цифрового фотоапарата. Зазвичай такі карти отримуються скануванням паперової карти з подальшою прив'язкою до деякої системи координат на поверхні Землі. Головні особливості таких карт - точна відповідність оригінальній карті (якщо не брати до уваги похибки сканування) і низька вартість виготовлення. До недоліків можна віднести порівняно великі обсяги графічного файлу мапи високої розподільчої здатності та неминучу втрату якості картинки при її збільшенні під час роботи з картою [3].

Векторні мапи складаються не з точок, а з об'єктів: річка, ліс, дорога, населений пункт, дорожній знак і багато іншого. Об'єкти описуються своїми характеристиками, як-то: розміри, положення на місцевості і іншими. Векторні мапи, на відміну від растрових, не мають характеристики розподільчої здатності. Це означає, що масштабування мапи відбувається без втрати якості картинки. Одна з переваг векторних карт - можливість онлайн-поновлення колекції карт, наприклад, через Інтернет [3].

Що стосується програмного забезпечення, в основному, це програма OziExplorer, яка дозволяє працювати з картами і планами місцевості на екрані комп'ютера. OziExplorer CE працює тільки з растровими картами і є навігаційним програмним забезпеченням для Pocket PC і Windows CE PDA. Опис можливостей, переваг і недоліків даної програми - це окрема тема.

2. ВИЗНАЧЕННЯ ПРОБЛЕМИ

Деяка будівельна компанія «N» зайнялася обліком пересування своєї будівельної техніки. Недорогим і результативним рішенням цієї проблеми стало забезпечення кожного транспорту GPS-реєстратором (GPS-реєстратор – невеликий пристрій, що фіксує координати і час пересування) з подальшим аналізом файлу маршрутів.

На рисунку 1 представлений фрагмент мапи Донецька і Донецької області (програма OziExplorer). На даному фрагменті немає зображення, будь-якого маршруту.

Для вилучення збереженої інформації з GPS-реєстратора використовується програма Wintec G_Tool. Це професійний програмний інструмент для обслуговування GPS-реєстраторів, в якому, в тому числі, передбачена можливість збереження інформації в

форматі OziExplorer (файл з розширенням * .plt). Даний файл має текстову систему кодування (проглядається програмою Блокнот). Вміст являє собою зведення координат і часу пересування транспорту. На рисунку 2 можна побачити фрагмент файлу OziExplorer_20100315_003_1.plt. Насправді даний файл формувався 18 днів і містить 23634 записів.

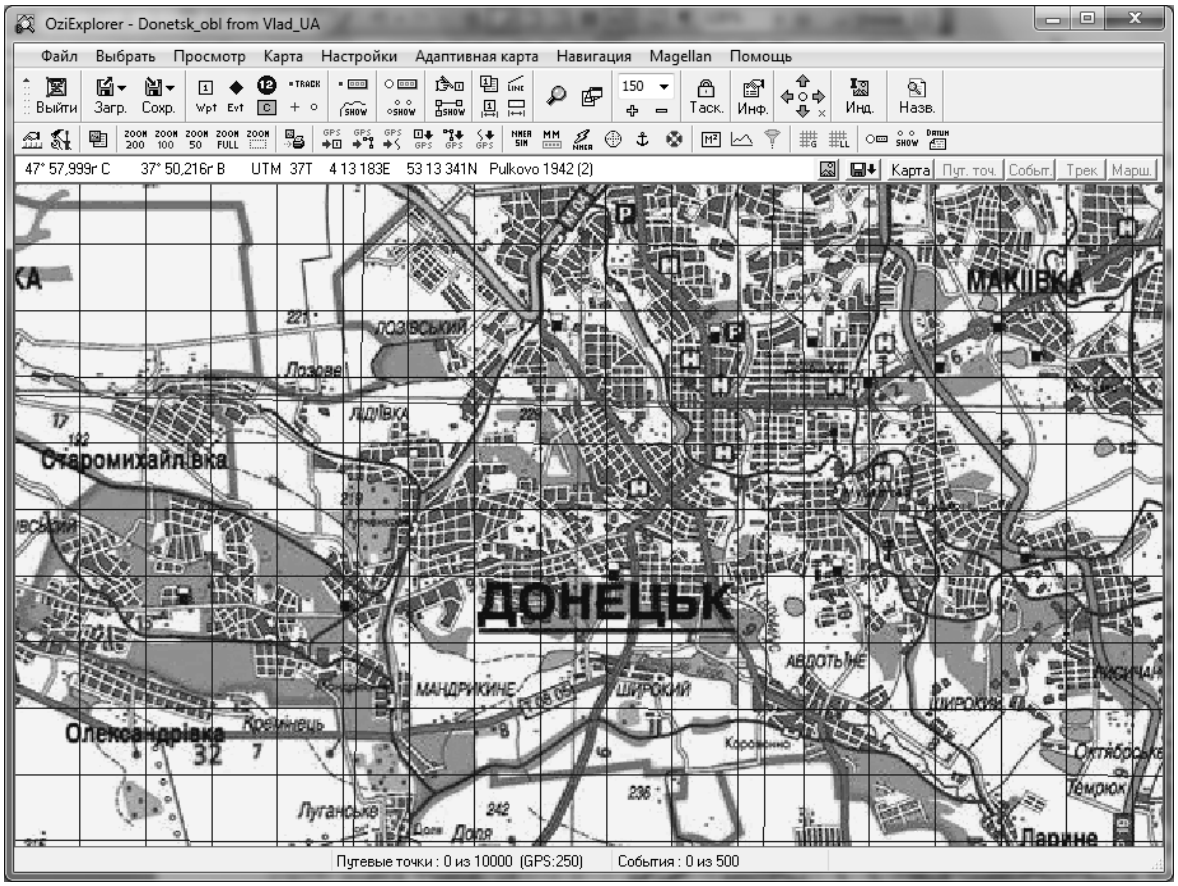


Рисунок 1

На рисунку 3 проілюстровано маршрут, інформація про який збережена у файлі OziExplorer_20100315_003_1.plt .

Зображення вісімнадцятиденного маршруту не дає чіткого уявлення про роботу тієї чи іншої бригади за вказаний період. Виникла необхідність вибірки координат і формування файлу добового пересування. Маючи графік роботи бригад, керівництво будівельної компанії змогло відстежити несанкціоноване використання техніки і витрату палива.

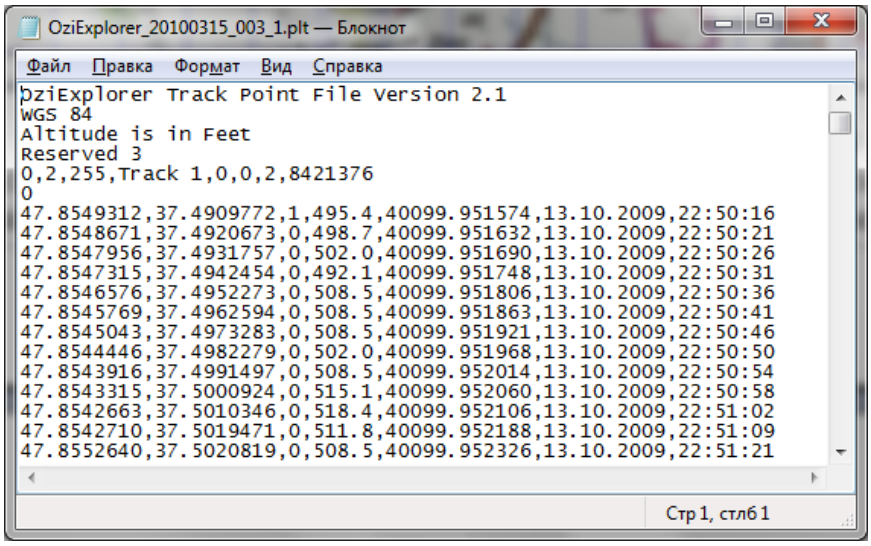


Рисунок 2

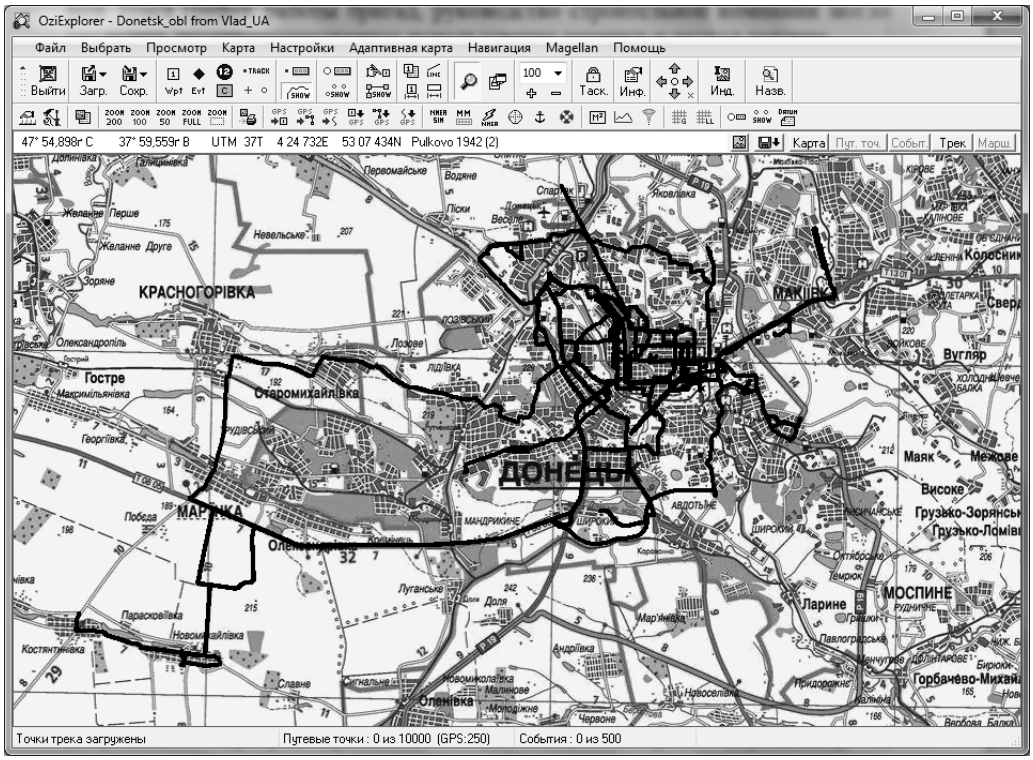


Рисунок 3

3. РОЗВ'ЯЗАННЯ ПРОБЛЕМИ.

Для вирішення поставленого завдання було розроблено програму Route. Інструмент програмування - Delphi 7, який має зручне середовище візуального програмування, компоненти якого (OpenDialog і SaveDialog) забезпечили простий і зрозумілий інтерфейс, а також широкий вибір процедур і функцій, призначених для обробки рядків і текстової інформації.

Використання даної програми дозволяє будь-якому користувачеві, маючи стандартні засоби діалогу, вказати файл, який є джерелом багатоденного маршруту, і записати результат у зазначеній папці. Результатом роботи програми є створення групи файлів у форматі OziExplorer (з розширенням * .plt). Інформація, що збережена в кожному з файлів, являє собою дані про добовий маршрут пересування.

При запуску програми з'являється вікно, що зображене на рисунку 4. У вікні «Открыть» (рисунок 5) користувач указує шлях до файлу з розширенням * .plt. У вікні «Сохранить как» (рисунок 6) користувачеві необхідно вказати папку для збереження результатів, в яку після закінчення роботи програми записується група файлів, що зберігають структуру та формат зрозумілий для відкриття в OziExplorer. Ім'я кожного створеного файлу задається автоматично і залежить від календарної дати.

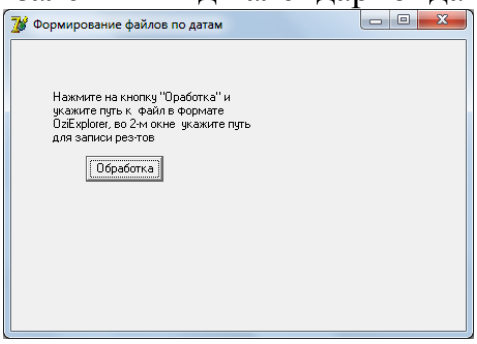


Рисунок 4

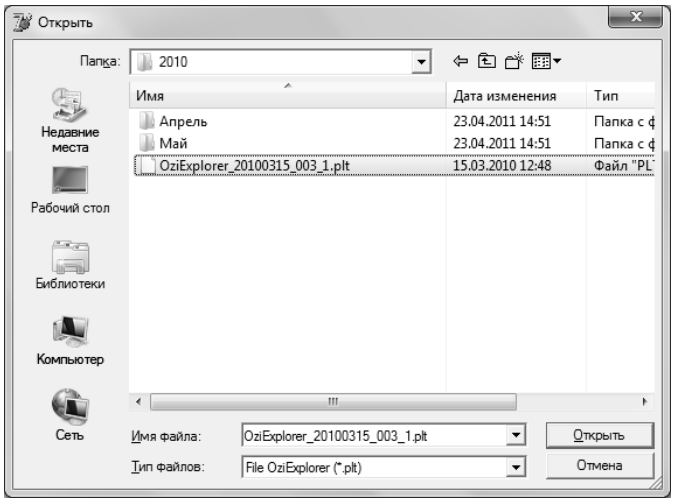


Рисунок 5

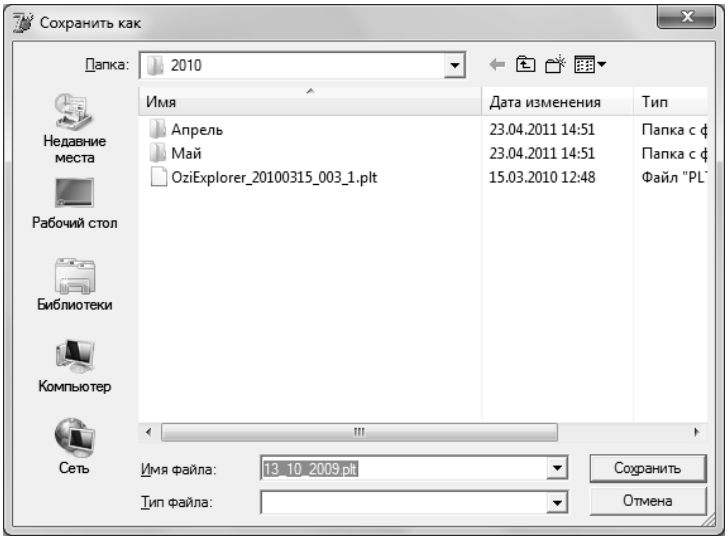


Рисунок 6

4. ВИСНОВОК

Після обробки програмою Route файлу OziExplorer_20100315_003_1.plt були отримані наступні файли – рисунок 7.

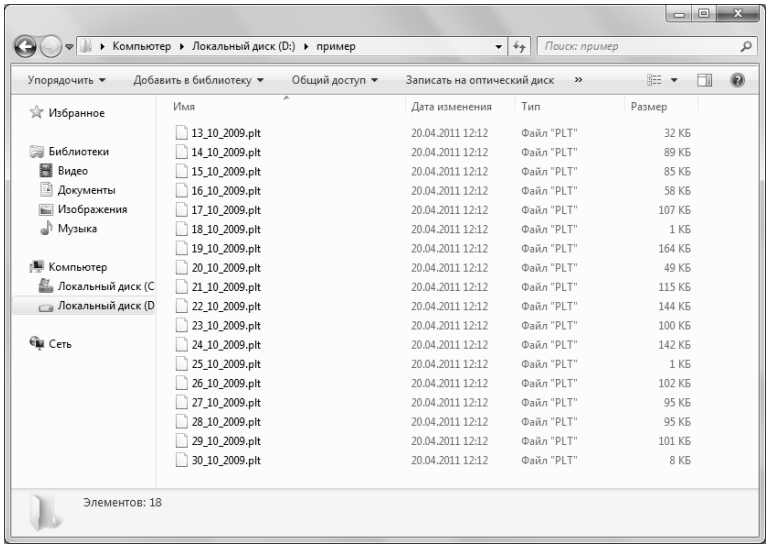


Рисунок 7

Приклад файлу з добовим обсягом даних, рисунок 8:

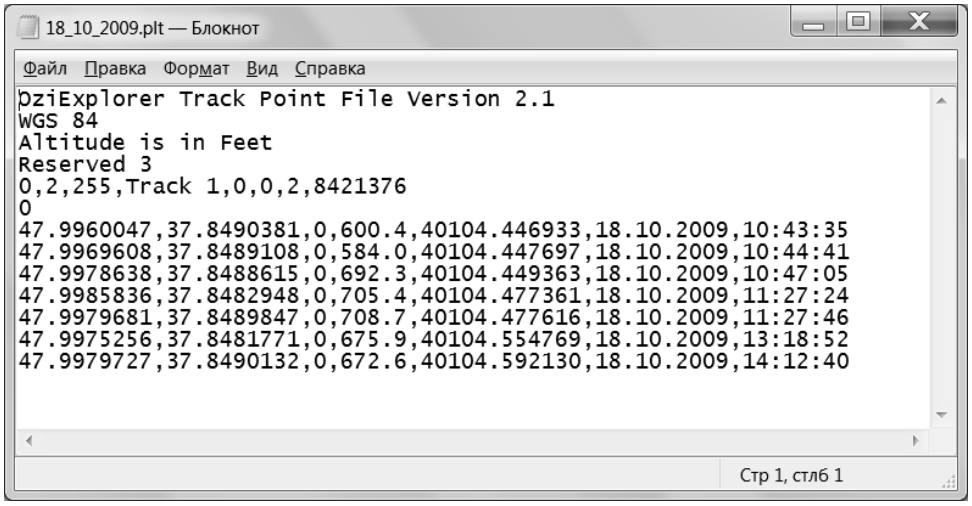


Рисунок 8

Файли, що були отримані, легко завантажуються в OziExplorer. І замість вісімнадцятиденного «вузла» можна спостерігати добове переміщення зазначеного транспорту, отримати інформацію про кілометраж і час. Наприклад, інформація за 20.09.2009, рисунок 9.

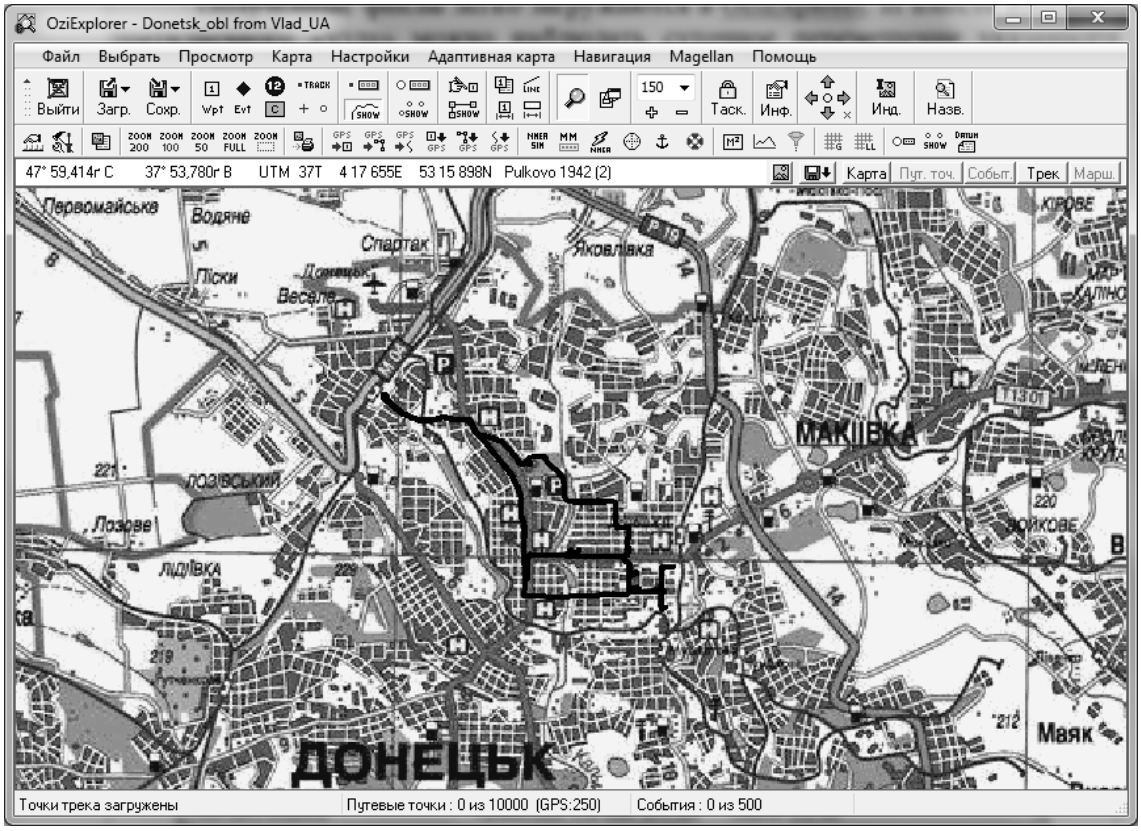


Рисунок 9

Таким чином, використовуючи сучасні засоби навігації (технічні та програмні), а також авторську програму Route, фірми, що займаються будівництвом, вантажоперевезеннями, пасажироперевезеннями і т.д., можуть відстежувати переміщення транспорту, що знаходиться у них на балансі.

ЛІТЕРАТУРА

[1] **Приемы работы с OziExplorer** [Електронний ресурс]. - Режим доступу: http://www.gps-profi.ru/Oziexplorer2.php

[2] **Спутниковый приёмник** [Електронний ресурс]. - Режим доступу: https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D1%83%D1%82%D0%BD%D0%B8%D0%BA%D0%BE%D0%B2%D1%8B%D0%B9_%D0%BF%D1%80%D0%B8%D1%91%D0%BC%D0%BD%D0%B8%D0%BA

[3] **GPS автомобильные навигаторы, трекеры, GPS карты, видеорегистраторы, эхолоты. | Приемы работы с OziExplorer**. [Електронний ресурс]. - Режим доступу: http://knifeclub.com.ua/forum/viewtopic.php?f=56&t=2483&start=40

Отримано 20.06.2017

УДК 004.274

АВТОНОМНА СИСТЕМА КЕРУВАННЯ ЕЛЕКТРОННИМИ ЗАМКАМИ НА ОСНОВІ NFC

В.С. Безбожний, С.О. Цололо

Донецький національний технічний університет, Україна

Розроблено автономну систему керування електронними замками на базі мікроконтролера ESP-12E з модулем зчитування NFC-міток. Система виконує керування відчиненням/зачиненням підключеного до неї електронного замка, на основі даних, зчитаних з NFC-карток відповідним модулем. До складу системи входить мікроконтролер з вбудованим мережевим модулем Wi-Fi ESP-12E, модуль NFC RC-522, реле перемикання стану електромеханічного замка та елемент живлення мікроконтролера.

Ключові слова: NFC, бездротові технології, модуляція, мікросхема, антена, приймач.

1. ВСТУП

«Near Field Communication» або «NFC» («зв'язок на невеликих відстанях») – технологія бездротового високочастотного зв'язку малого радіусу дії. Ця технологія дає можливість обміну даними між пристроями, насамперед смартфонами та безконтактними платіжними терміналами, що знаходяться на відстані близько 10 см [1]. NFC є підмножиною в сімействі RFID-технології. Зокрема, NFC є гілкою високочастотної RFID, і працює на частоті 13,56 МГц. NFC розроблялася фактично у якості безпечної форми обміну даними, а пристрій може бути як NFC-зчитувачем, так і NFC-міткою [2].

За визначенням, RFID – це метод однозначної ідентифікації предметів за допомогою радіохвиль. Як мінімум, система RFID містить тег, зчитувач і антену. Принцип роботи полягає у тому, що зчитувач посилає сигнал до опитувальної мітки через антену, а мітка відповідає з її унікальною інформацією. [3]. Технологія дає можливість обміну даними між пристроями, насамперед смартфонами та безконтактними платіжними терміналами, що знаходяться на відстані близько 10 см, та не виключає можливості використання спеціальних міток у вигляді карток, брелків або наліпок.

Існує декілька критеріїв класифікації NFC-міток (рис. 1) – за робочою частотою, за джерелом живлення, за типом пам'яті.

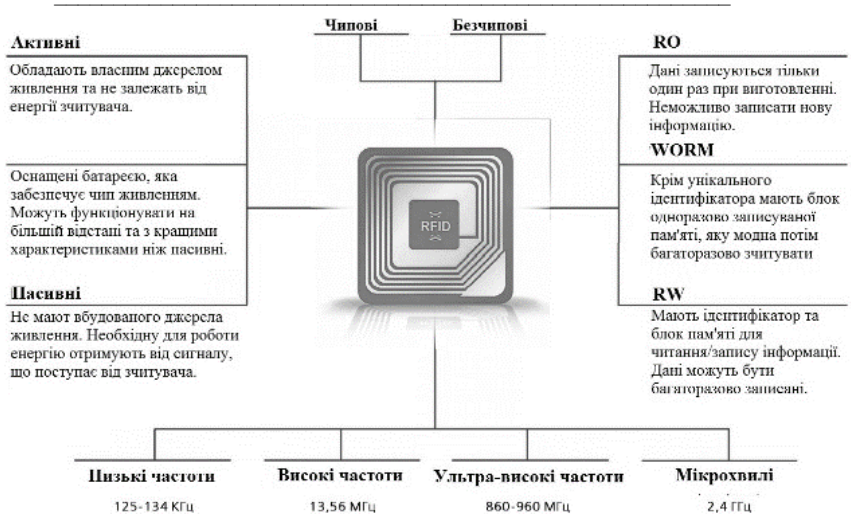


Рисунок 1 – Класифікація NFC-міток

За джерелом живлення NFC-мітки діляться на пасивні, активні, та напівпасивні. Пасивні мітки живляться за рахунок електричного струму, індукованого в антені електромагнітним сигналом від зчитувача, що забезпечує достатню потужність для функціонування кремнієвого CMOS-чипу, розміщеного в мітці, та передачі відповідного сигналу [3].

Пасивні мітки ВЧ-діапазону передають сигнал методом модуляції навантаження сигналу несучої частоти. Кожна мітка має ідентифікований номер. Пасивні мітки можуть містити пам'ять EEPROM-типу. Дальність дії міток складає 1-200 см (ВЧ-мітки) та 1-10 метрів (УВЧ-та СВЧ мітки) [4].

Активні RFID-мітки мають власне джерело живлення та не залежать від енергії зчитувача, тому вони зчитуються на більшій відстані, мають більші розміри та можуть бути оснащені додатковою електронікою. Серед недоліків активних міток слід відзначити дороговизну їх виготовлення та малу мобільність внаслідок обмежень ємності батарей. Активні мітки мають набагато більший радіус зчитування (до 300 м) та об'єм пам'яті, ніж пасивні.

Напівпасивні RFID-мітки дуже схожі на пасивні, але оснащені батареєю, яка забезпечує чип енергоживленням.

В основі роботи NFC лежить індуктивний зв'язок (рис. 2). Частота роботи такої системи складає – 13,56 МГц (HF RFID). Сигнал піддається амплітудній маніпуляції OOK з різною глибиною 100% або 10% і фазової маніпуляції BPSK. При передачі інформації пасивного пристрою використовується амплітудна маніпуляція ASK. Кожен пристрій має власне джерело живлення, тому сигнал несучої відключається відразу після закінчення передачі.

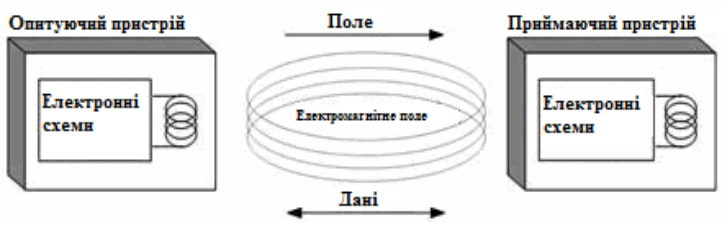


Рисунок 2 – Принцип обміну в NFC

За рахунок індуктивного зв'язку між опитуваних і прослуховуючих пристроями пасивне пристрій впливає на активну. Зміна імпедансу прослуховувального пристрою викликає зміну амплітуди або фази напруги на антені опитувального пристрою, який він виявляє. Цей механізм називається модуляцією навантаження. Вона виконується в режимі прослуховування із застосуванням допоміжної несучої 848 кГц. Залежно від стандарту застосовується амплітудна або фазова маніпуляція [4].

2. СТРУКТУРА СИСТЕМИ

До автономної систему керування електронними замками, що пропонується в роботі, від самого початку були висунута вимога функціонування від автономного джерела живлення та відносно невелика вартість.

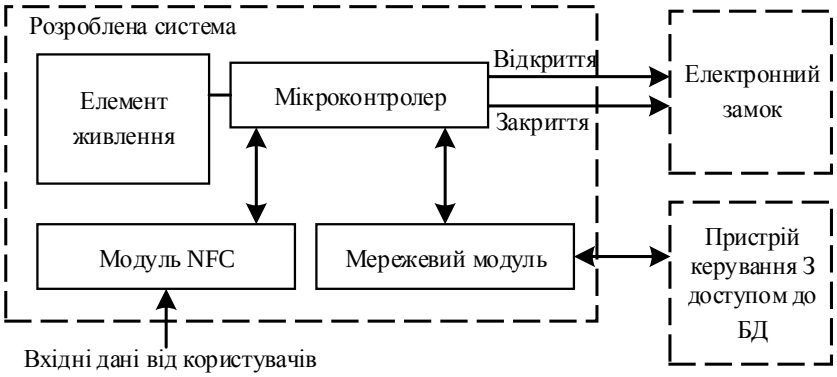


Рисунок 3 – Структурна схема системи

Відповідно до вимог система складається з модулів (рис. 3):

1. *Мікроконтролер* – керує модулями системи та обробки зовнішніх даних.
2. *Модуль зчитування міток* – зчитування ідентифікаційних міток з карток-перепусток або мобільних телефонів.

3. *Мережевий модуль* – зв'язок системи із базою даних або віддалене керування системою.
4. *Елемент живлення* – автономна робота системи протягом тривалого часу.

3. РЕАЛІЗАЦІЯ СИСТЕМИ

Система повинна бути автономною та працювати тривалий час від елементу живлення та мати відносно малу вартість. Було розглянуто декілька варіантів мікроконтролерів – Arduino Uno R3 з модулем Wi-Fi, МК серії STM з модулем Wi-Fi, Wi-Fi-модуль ESP-12E, з власним обчислювальним елементом [6].

Найбільш поширеними та простими у програмуванні є мікроконтролери Arduino. Але, оскільки модуль ESP-12E може виконувати ті ж самі функції, що і STM та Arduino, і вже має вбудований Wi-Fi модуль, що дасть змогу підключення ПК до локальної мережі через даний модуль, та його вартість не перевищує інші варіанти було вирішено використовувати ESP-12E [7].

Серед тих модулів NFC, що можна придбати та протестувати власноруч, слід відмітити наступні:

- RFID-RC522 – дешевий та універсальний модуль, сумісний з найбільш поширеними RFID-мітками та протоколами;
- PN532 – модуль на базі мікросхеми NXP PN532;
- SL060 – модуль, розроблений на базі ретранслятора NXPNFCIC, що є читаючим/записуючим пристроєм.

Оскільки модуль RC522 має найбільший діапазон міток, що підтримуються, та підтримує декілька режимів підключення було вирішено використовувати саме його.

У якості актуатора для керування електронним замком було використано реле KY-019 – 1-канальный модуль, що має один нормально-замкнутий та один нормально-розімкнутий контакти [8]. Оскільки деякі моделі електронних замків мають лінію керування не тільки для відчинення а й для зачинення дверей то було вирішено оснастити систему двома реле для реалізації відчинення та зачинення.

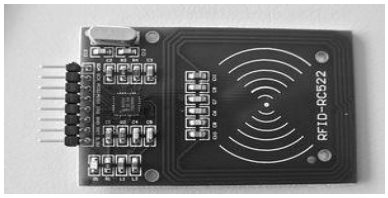
Зважаючи на те, що мікроконтролер може живитися від звичайного USB-інтерфейсу було вирішено використати портативний powerbank невеликої ємності (в межах 2000 мАг). Такий вибір зумовлено тим, що система орієнтована на підключення до електронних замків вже обладнаних власним джерелом живлення, достатнім для відчинення/зачинення, керування подачею напруги від яких буде займатися розроблена система.

Отже система складається з чотирьох компонентів:

1. *Мікроконтролер ESP-12E* (рис. 4а) – головна частина системи, яка виконує всі обчислення та керування відчиненням/зачиненням дверей.
2. *Модуль RFID-RC522* (рис. 4б) – зчитує персональні дані робітників з персоніфікованих перепусток, що дозволяє однозначно ідентифікувати кожного робітника.
3. *Реле KY-019* (рис. 4в) – перемикає стан електронного замка сигналами від системи ідентифікації.
4. Trust 2000 mAh (рис. 4г) – елемент живлення системи.



а) модуль ESP-12E



б) модуль RFID-RC522



в) реле KY-019



г) елемент живлення Trust 2000 mAh

Рисунок 4 – Компоненти системи

4. ВИСНОВКИ

В роботі запропонована автономна системи керування електронними замками на базі технології NFC. Розроблена система реалізує такі основні функції:

- зчитування даних карток-перепусток та ідентифікація користувачів;
- дистанційне керування відчиненням/зачиненням замка за допомогою мережевого модуля;
- внесення записів до бази даних про відчинення-зачинення дверей за допомогою мережевого модулю.

Система реалізована на модульній основі. Функціональні вузли можуть бути замінені на подібні з мінімальними змінами у програмній частині. Джерело живлення може бути з легкістю заряджене без вилучення із системи або замінене на аналогічне у разі непридатності до роботи.

На даний момент орієнтовна вартість системи складає:

- мікроконтролер ESP-12E – 10$;
- NFC-модуль RC-522 – 2,5$;
- 2 реле відчинення та зачинення замка KY-019 – 1$;
- PowerBank 2000 mAh для живлення системи – 5$;
- витратні матеріали – 2.5$.

Отже загальна вартість реалізованої системи складає приблизно 21$, що є набагато менше за аналогічні серійні системи. Наприклад, система ZKTeco K300 коштує приблизно 250$. Враховуючи середню вартість електронних замків, яка є в межах 100$, та орієнтовну вартість розробленої системи, що буде керувати його роботою, 21$ то все рівно вартість готового рішення є вдвічі більшою за розроблений прототип в поєднанні з серійним замком та є безперечною перевагою для невеликих підприємств та державних установ.

ЛІТЕРАТУРА

[1] "Information technology - Telecommunications and information exchange between systems — Near Field Communication — Interface and Protocol (NFCIP-1)", ISO/IEC 18092, First Edition, 2004-04-01.

[2] Klaus Finkenzeller, "RFID Handbuch", Hanser Verlag, 2002.

[3] Morris Dworkin, "Recommendation for Block Cipher Modes of Operation", NIST Special Publication 800-38A, 2001.

[4] Security Concerns with NFC Technology [Електронний ресурс]. – Режим доступу: http://www.nearfieldcommunication.org

[5] Технология NFC в смартфонах и ее практическое использование [Електронный ресурс] – Режим доступу: http://www.ixbt.com/mobile/nfc-2013.shtml

[6] ESP-12E WiFi Module [електронний ресурс]. – Режим доступу: https://mintbox.in/media/esp-12e.pdf

[7] Безбожний В.С., Цололо С.О. Розробка автоматизованої системи обліку робочого часу на базі технології NFC // «ТАК»: телекомунікації, автоматика, комп'ютерно-інтегровані технології: зб. доповідей Всеукр. наук.-практ. конф. молодих вчених, 29-30 листопада 2016 р. / ДВНЗ «ДонНТУ – Покровськ: ДВНЗ «ДонНТУ», 2016. – С. 113-115

[8] Arduino KY-019 5V relay module [електронний ресурс]. – Режим доступу: https://tkkrlab.nl/wiki/Arduino_KY-019_5V_relay_module

Отримано 27.06.2017

УДК 004.274

ПЛАТФОРМА РОЗУМНОГО БУДИНКУ НА ОСНОВІ ARDUINO MEGA 2560

В.Г. Новіков, С.О. Цололо

Донецький національний технічний університет, Україна

Розроблено платформу розумного будинку з мережевим доступом на основі Arduino Mega 2560, що вирішує комплекс завдань, пов'язаних з автоматизацією та моніторингом основних електричних вузлів за допомогою інтернету. Пропонуються мінімальні витрати на виробництво і обслуговування з можливістю розширення системи, що складається з центрального керуючого модуля, керованих модулів – датчиків та актуаторів, і веб-інтерфейсу.

Ключові слова: розумний будинок, актуатор, датчик, керування, моніторинг.

1. ВСТУП

У наш час все більшу і більшу популярність завойовують системи, спрямовані на комфорт, енергозбереження та економію коштів. Однією з таких систем є система "розумний будинок". Система розумного будинку складається з двох основних частин: апаратної та програмної. Апаратна частина складається з центрального сервера і спеціальних контролерів, кожен з яких виконує окрему функцію. Наприклад, релейний модуль управляє шторами і воротами, інфрачервоний модуль управляє аудіо/відео апаратурою тощо. Усіма контролерами управляє центральний сервер за командами від панелей управління. Вся логіка управління закладається в центральний сервер [1].

Принципових відмінностей в залежності від масштабу і площі будівлі немає, система легко масштабується і розширюється. Єдиним рішенням для управління будівлею більшої площі є крок в сторону збільшення продуктивної потужності серверного обладнання. Комплектація обладнання змінюється несуттєво, технічні схеми залишаються незмінними. Принципові відмінності між автоматизацією житлових і офісних будівель помітні в основному тільки на етапі проектування. Принципово можуть відрізнятися технічні рішення, але не сама технологія. З точки зору комплектації устаткування принципових відмінностей немає. З точки зору функціональної наповнюваності відмінності несуттєві. Отже, основною метою роботи є створення платформи домашньої

автоматизації з мережевим доступом на основі мікроконтролера Arduino Mega.

Задачі роботи:

- аналіз необхідної функціональності системи;
- оцінка можливості управління за допомогою інтернету;
- дослідження роботи бездротових модулів обміну даними всередині системи;
- оцінка можливості масштабованості системи за рахунок бездротових модулів.

2. СТРУКТУРА ПЛАТФОРМИ

Структурно платформа розумного будинку складається з трьох основних елементів (рис. 1).

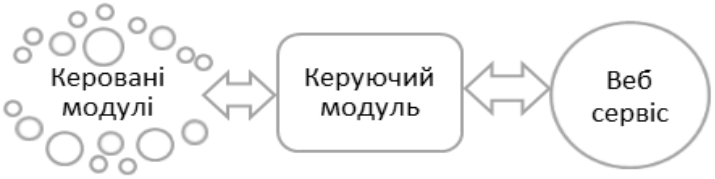


Рисунок 1 – Загальна структура платформи

Керовані модулі – датчики і актуатори. Датчики вимірюють стан приміщення на поточний момент: температура, вологість, тиск, напруга в мережі тощо і передають отримані дані на керуючий модуль. Актуатори – прилади, здатні управляти подачею напруги, змінювати стан електропобутової техніки і фізичний стан деяких об'єктів (наприклад відкриття жалюзі), відповідно до команд, отриманими від керуючого модуля.

Керуючий модуль приймає дані з датчиків, обробляє їх, і відповідно до заданих сценаріїв, виконує управління актуаторами. Так само веде двосторонній обмін даними з сервером, передаючи отримані дані і беручи подальші інструкції.

Веб-сервіс візуалізує отриману інформацію від керуючого модуля в зрозумілій для людини формі і дозволяє змінювати прописані інструкції керуючого модуля.

Розглянемо загальну структуру платформи розумного будинку більш конкретно (рис. 2).

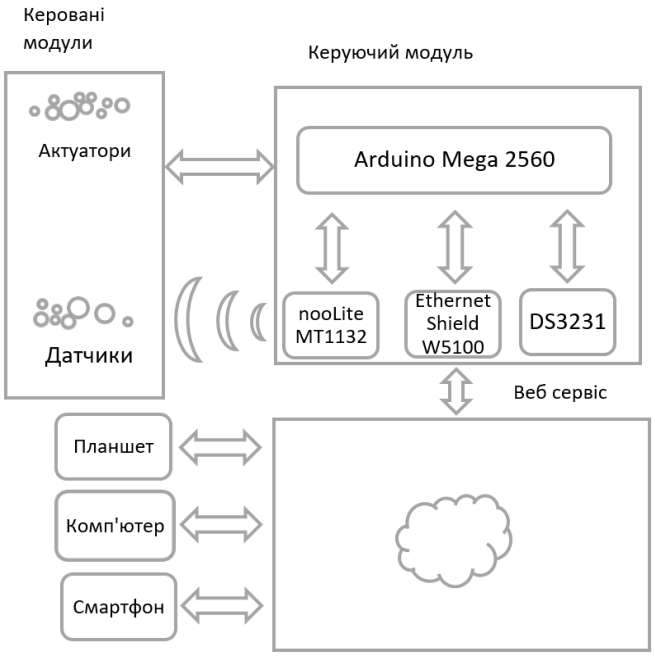


Рисунок 2 – Детальна структура платформи

Провідні датчики і актуатори підключаються безпосередньо до МК, бездротові – завдяки схемі nooLite MT1132, яка в свою чергу з'єднана із Arduino. Годинник реального часу DS3231 передає дані про поточну дату і час на МК, що необхідні для автономного управління світлом. Віддалене керування здійснюється за допомогою телефону, планшета або комп'ютера через веб-сервіс. Arduino зчитує і передає дані на веб-сервіс завдяки Ethernet Shield W5100 [6].

3. КЕРУЮЧИЙ МОДУЛЬ

Основою модуля є *мікроконтролер Arduino Mega 2560*. У його склад входить все необхідне для реалізації вбудованих систем: 54 цифрових входів/виходів (з яких 15 можуть використовуватися в якості ШІМ-виходів), 16 аналогових входів, 4 UART (апаратних приймача для реалізації послідовних інтерфейсів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм ICSP для внутрисхемного програмування і кнопка скидання. Для початку роботи з пристроєм досить просто подати живлення від AC/DC-адаптера або батарейки, або підключити його до комп'ютера за допомогою USB-кабелю [8].

Блок Ethernet Shield W5100 забезпечує зв'язок центрального модуля з веб-інтерфейсом. Плата побудована на базі Ethernet-контролера Wiznet W5100, який підтримує стек мережевих протоколів (IP) і дозволяє працювати як з TCP-, так і з UDP-протоколами. При

цьому мікросхема може обслуговувати до чотирьох одночасно відкритих сокет-з'єднань. Для підключення плати розширення до Arduino передбачений спеціальний роз'єм, який представляє собою металеві висновки ("папа") з одного боку плати та гнізда ("мама") – з іншого боку. Така конструкція дозволяє підключити до Arduino відразу кілька плат розширення, розмістивши їх одну над іншою. На платі передбачений роз'єм для підключення micro-SD карти пам'яті, що дає можливість зберігання файлів і організації мережевого доступу до них – це дуже зручно для нашого випадку, так як саме на карті будуть файли інтернет-ресурсу [7].

Для прив'язки до часу використовується *мікросхема DS3231* – годинник реального часу з вбудованим кварцевим резонатором із температурною компенсацією. Інтерфейс передачі даних – I2C. У цій мікросхемі є також вхід для напруги резервної батареї, при відключенні основного живлення мікросхема автоматично перемикається на роботу від резервної батареї, точність ходу від резервної батареї не порушується. В DS3231 підтримується підрахунок секунд, хвилин, годин, днів місяця (дати), днів тижня, місяців і років (з урахуванням високосного року для місяців) [3].

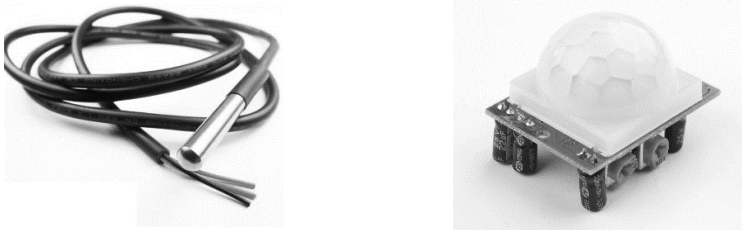
Модуль передавача nooLite MT1132 призначений для бездротового управління силовими блоками системи nooLite з платформи Arduino, мікроконтролерів або ПК. Конструктивно модуль виконаний у вигляді друкованої плати, на якій розташований керуючий контролер і передавач. Управління модулем здійснюється через послідовний інтерфейс UART. Прийняті за команди управління модуль передає на силові блоки через вбудований радіопередавач [9].

4. КЕРОВАНІ МОДУЛІ

Цифровий датчик температури Dallas 18B20 (DS18B20) є компактним, точним і недорогим цифровим датчиком температури (рис. 3а). Датчик використовує інтерфейс 1-Wire, розроблений компанією Dallas. Цей інтерфейс є широко поширений і досить простим в освоєнні. Додатковим його плюсом є можливість паралельного підключення декількох датчиків на одну шину даних [4].

Інфрачервоний датчик руху HC-SR501 (рис. 3б) використовується для виявлення в контрольованій зоні руху об'єктів, які випромінюють інфрачервоне випромінювання (тепло). Принцип роботи датчика заснований на піроелектриці – властивості генерувати певне електричне поле при опроміненні матеріалу інфрачервоними (тепловими) променями [5]. Над чутливим елементом встановлена

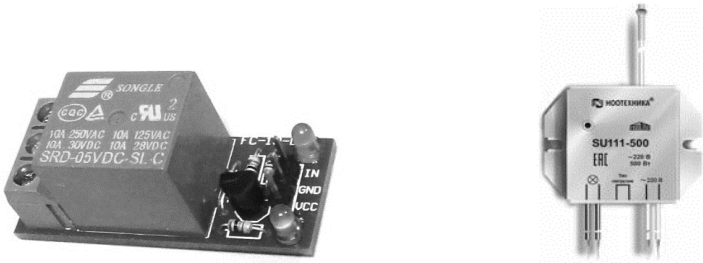
лінза Френеля, яка використовується для збільшення радіусу огляду та посилення вхідного інфрачервоного сигналу.



а) *б)*

Рисунок 3 – Датчик DS18B20 (а) та HC-SR501 (б)

Типовим актуатором в системах автоматизації служить блок реле. В роботі використовується блок на основі реле фірми SONGLE SRD-05VDC (рис. 4а). Дане реле управляється напругою 5V і здатне комутувати до 10A 30V DC і 10A 250V AC [10].



а) *б)*

Рисунок 4 – Блок реле SONGLE SRD-05VDC (а) та силовий блок SU111-500 (б)

Блок SU111-500 (рис. 4б) – універсальний радіовимикач, призначений для дистанційного включення/вимикання освітлення з можливістю регулювання яскравості для діміруємих джерел світла (лампи розжарювання, світлодіодні діміруємні лампи). Блок може працювати в двох режимах: релейному і дімірованому, що залежить від передбачуваного типу навантаження. У релейному режимі SU111-500 здатний працювати з будь-яким типом навантаження [9].

5. ВИСНОВКИ

Розроблена платформа реалізує моніторинг стану житлового приміщення та енергоспоживання в поточний момент часу, управління по заздалегідь заданих сценаріях, а також віддалений доступ до системи та управління основними мережними вузлами.

Система залишає можливість додавання нових функціональних вузлів без заміни основного модуля. Перехід на дану систему дозволяє

істотно знизити матеріальні витрати на закупівлю обладнання, його подальше обслуговування і залишає можливість вирішення додаткових завдань, пов'язаних з віддаленим моніторингом, сигналізацією і управлінням.

Практична значимість проекту полягає в зниженні матеріальних витрат на розробку і виготовлення. На даний момент вартість обладнання наступна:

- Arduino Mega 2560 – 47 $;
- Ethernet Shield W5100 – 7 $;
- DS3231 – 1,64 $;
- nooLite MT1132 – 17,8 $;
- SU111-500 – 23.9 $ x2;
- блок реле – 1,52 $ x5;
- DS18B20 – 1,95$ x3;
- HC-SR501 – 1,25$ x5;
- витратні матеріали - 2.5 $.

Разом на сьогоднішній день загальна вартість базового варіанту системи становить близько 145 $. У аналогічних комплексних рішень, які на даний момент представлені на світовому ринку і ринку СНД зокрема, роздрібна ціна знаходиться в межах від 2000 до 10000 € + близько 30-40% вартості додатково за послуги монтажу та запуску системи в експлуатацію.

ЛІТЕРАТУРА

[1] Система Умный дом [електронний ресурс]. – Режим доступу: http://www.unicc.kiev.ua/articles/sistema_umnii_dom/.

[2] Что такое «Умный дом»? Принципы построения [електронний ресурс]. – Режим доступу: http://shkolazhizni.ru/family/articles/65256/

[3] Гололобов В. С чего начинаются роботы? – М.: Додэка-XX1, 2011. – 189 с.

[4] Иго Т. Arduino, датчики и сети для связи устройств, 2-е изд. – СПб.: БХВ – Петербург, 2015. – 544с.

[5] Блум Джереми Изучаем Arduino: инструменты и методы технического волшебства, пер с англ. – СПб: БХВ – Петербург, 2015. – 336с.

[6] Соммер У. Программирование микроконтроллерных плат Arduino/Freeduino (2-е изд.) пер с нем. – СПб: БХВ – Петербург, 2016. – 256с.

[7] Петин В. Датчики для Arduino и Raspberry Pi в проектах Internet of Things – СПб: БХВ – Петербург, 2016. – 320с.

[8] Arduino Mega 2560 [електронний ресурс]. – Режим доступу: http://arduino.ua/ru/hardware/Mega2560/.

[9] Официальный сайт «НООТЕХНИКА» [електронний ресурс]. – Режим доступу: http://www.noo.com.by.

Отримано 26.06.2017

НАУКОВЕ ВИДАННЯ

М А Т Е Р І А Л И

Сьомої міжнародної науково-технічної конференції

"МОДЕЛЮВАННЯ І КОМП'ЮТЕРНА ГРАФІКА"

(українською, російською, англійською мовами)

Формат $64\times80^{1}/_{16}$
Ум. друк. арк. 11,9
Тираж 80 прим. Замовлення № 0911

Видавець: Державний вищий навчальний заклад «Донецький національний технічний університет», пл. Шибанкова, 2, м. Покровськ, 85300, Україна

Свідоцтво про державну реєстрацію суб'єкта видавничої справи: ДК 4911 від 09.06.2015.

Надруковано: Краматорська типографія, м. Краматорськ, вул. Паркова, 27