



СЕКЦИЯ 7

**МОДЕЛИРОВАНИЕ И
СОВРЕМЕННЫЕ
КОМПЬЮТЕРНЫЕ
ТЕХНОЛОГИИ**

УДК 330.42

ВЛИЯНИЕ ШОКОВЫХ ВОЗДЕЙСТВИЙ НА ФОНДОВЫЙ ИНДЕКС**Андриенко В.М., Богданова В.А., Спиваков О.Г.**

Одесский национальный политехнический университет

Кафедра информационных систем в менеджменте

E-mail:andrienko.v@gmail.com

Аннотация:

Андриенко В.М., Богданова В.А., Спиваков О.Г. Влияние шоковых воздействий на фондовый индекс. Работа посвящена анализу и прогнозированию важнейшего макроэкономического показателя Украины – индексу фондового рынка ПФТС. Применен подход к идентификации динамики индекса с точки зрения продолжительности присутствия в ней последствий возмущений, вызванных некоторым внешним шоковым воздействием.

Общая постановка проблемы

Состояние фондового рынка играет важную роль для стабильного развития экономики. Крах фондового рынка, то есть сильное падение (шок) курсовой стоимости ценных бумаг за короткий промежуток времени, может вызвать спад и депрессию в экономике. Принимая решения об инвестициях, финансовый менеджер постоянно оценивает поведение в будущем, как отдельных финансовых активов, так и рынка в целом. Участнику рынка нужно хотя бы приблизительно представлять картину будущего. Именно поэтому на первый план выдвигается задача оценки состояния и тенденции развития ситуаций на фондовом рынке. С этой целью вся текущая и прошлая финансовая информация тщательно анализируется с помощью методов финансового анализа. Это дает понимание прошлого и текущего состояния рынка. Однако каким бы детальным не было это понимание, прогнозы, составленные только на такого рода анализе, не могут служить надежной основой для принятия решений об инвестировании. В этой связи, построение математических моделей, позволяющих лучше понять структуру и поведение рынка как единого целого, так и его составляющих, долгое время привлекали и продолжают привлекать внимание исследователей и практиков. Проблема моделирования динамики цен рыночных активов и их прогнозирования является достаточно сложной, и ее нельзя назвать в настоящее время решенной.

Исследования

1. *Индекс украинского фондового рынка ПФТС.* Основными индикаторами фондового рынка являются индексы, рассчитываемые на основании котировок определенной группы ценных бумаг. В зависимости от того, какие компоненты содержит индекс, он может отражать поведение определенной группы акций или всего фондового рынка. Индекс Украинского фондового рынка ПФТС (первая фондовая торговая система) признан международной финансовой корпорацией (МФК) как единственный индекс, используемый этой организацией при мониторинге внутреннего состояния украинского фондового рынка. Индекс определяет средний уровень наиболее ликвидных украинских акций, которые имеют наибольшую рыночную капитализацию. Информацию о состоянии индекса ПФТС публикует на своем сайте «Первая фондовая торговая система», которая представляет собой электронную биржу ценных бумаг Украины и поддерживает работу национальной электронной системы торговли ценными бумагами в режиме «online».

2. *Математико-статистические методы моделирования временных рядов.*

Большинство математико-статистических методов имеет дело с моделями, в которых наблюдения предполагаются независимыми и одинаково распределенными. При этом

основное внимание уделяется проблемам идентификации моделей, отбору эндогенных и экзогенных показателей, но почти не обращается внимания на формальный анализ структуры исходных статистических рядов. Зависимость между наблюдениями чаще всего рассматривается как помеха в эффективном применении этих методов. Однако разнообразные данные в экономике, социологии, финансах, коммерции и других сферах человеческой деятельности поступают в форме *временных рядов*, в которых наблюдения взаимно зависимы, и характер этой зависимости как раз и представляет главный интерес для исследователя. *Временным рядом* называется совокупность наблюдений экономического показателя в различные моменты времени. Обычно временной ряд рассматривают как выборку из последовательности случайных величин X_t , где t принимает целочисленные значения от 1 до T . Совокупность случайных величин $\{X_t, t \in [1, T]\}$ называют *дискретным случайным процессом* или *стохастическим процессом*. Принципиальные отличия временного ряда от последовательности наблюдений, образующих случайную выборку, заключаются в следующем:

- в отличие от элементов случайной выборки члены временного ряда не являются независимыми;
- члены временного ряда не обязательно являются одинаково распределенными.

Это означает, что свойства и правила статистического анализа случайной выборки нельзя распространять на временные ряды. С другой стороны, взаимозависимость членов временного ряда создает свою специфическую базу для построения прогнозных значений анализируемого показателя по наблюдаемым значениям.

В 1938 году Вольд доказал следующий фундаментальный результат [1]. Чисто недетерминированный стационарный в широком смысле [2] случайный процесс X_t может быть представлен в следующем виде: $X_t - \bar{X}_t = \sum_{\tau=0}^{\infty} \psi_{\tau} \varepsilon_{t-\tau}$, где \bar{X}_t - математическое ожидание этого процесса, а ε_t - белый шум с конечными математическим ожиданием и дисперсией. То есть всякий стационарный в широком смысле случайный процесс представляется в виде линейной комбинации белых шумов. При этом должно выполняться условие $\sum_{\tau=0}^{\infty} |\psi_{\tau}| < \infty$.

Поскольку реализации белого шума не наблюдаемы, весовые коэффициенты определены с точностью до множителя. Без потери общности можно считать, что $\psi_0 = 1$. Чем больше весовой коэффициент ψ_{τ} , тем больше влияние случайного возмущений в момент $t - \tau$ на текущий момент t . Оказалось, что во многих случаях достаточно рассматривать не общее представление Вольда, а его частные случаи, когда число слагаемых конечно. Такими частными случаями являются популярные в эконометрике авторегрессионные модели $AR(p)$, $MA(q)$, $ARMA(p, q)$.

Экономические показатели, в частности показатели фондовых рынков, не всегда ведут себя стационарным образом. Из макроэкономики известно их *сезонное* и *циклическое* поведение, кроме того, они могут иметь *тренд*. Часто эти виды компонент присутствуют в ряде одновременно.

Аналитически временной ряд можно выразить уравнением вида:

$$X(t) = f(t) + S(t) + \varepsilon(t), \quad (1)$$

где $f(t)$ - тренд (долговременная тенденция) развития;

$S(t)$ - сезонная (периодическая) компонента;

$\varepsilon(t)$ - случайная величина (случайный компонент).

Функция $f(t)$ определяет общую тенденцию развития изучаемого явления. Тренд может

быть выражен как детерминированной, так и случайной функциями, либо их комбинацией. Компоненты временного ряда $f(t)$, $S(t)$ и $\varepsilon(t)$ ненаблюдаемы. Они являются теоретическими величинами. В рядах с детерминированным трендом влияние предыдущих шоковых воздействий (резких изменений) затухает с течением времени, а в рядах со стохастическим трендом такое затухание отсутствует, и каждый отдельный шок влияет с одинаковой силой на все последующие значения ряда. Детерминированные компоненты $f(t)$ и $S(t)$ обнаруживаются с помощью спектрального анализа и их необходимо исключить. Детерминированная составляющая имеет неограниченную спектральную плотность на низких частотах, а при наличии периодических составляющих спектральная плотность имеет максимумы (пики). Бокс и Дженкинс [3] предложили компоненту $f(t)$ исключать дифференцированием, то есть вычислением последовательных разностей. Если ряд после вычисления d последовательных разностей приводится к стационарному, то такой ряд называют $ARIMA(p,d,q)$ (авторегрессионные проинтегрированные скользящего среднего – *Autoregressive Integrated Moving Average*) порядка (p,d,q) , которые моделируют различные ситуации, встречающиеся при анализе стационарных и нестационарных рядов.. При этом p параметр AR - части, d – степень интеграции, q – параметр MA - части. Периодическую составляющую удаляют вычитанием ее из ряда. Стохастический тренд обнаруживается с помощью автокорреляционного анализа. Автокорреляционная функция в этом случае медленно убывает. Такие ряды называют «временные ряды с долговременной корреляционной зависимостью (*time series with long memory*)». В работах зарубежных ученых, в первую очередь, *C.W.Granger, J.R.Hosking, P.M.Robinson, R. Beran* [4], был предложен новый класс моделей $ARFIMA(p,d,q)$, допускающий возможность нецелого параметра d и получивший название авторегрессионный дробно - интегрированный процесс скользящего среднего. Характеристики таких временных рядов обладают важными свойствами, например, X_t является стационарным и обратимым при $d \in (-1/2, 1/2)$. При этом X_t можно представить в виде [5]:

$$\Phi(B)(1-B)^d X_t = \Theta(B)\varepsilon_t \quad (2)$$

де $B = X_{t-1} / X_t$ - оператор сдвига назад, $\Phi(B) = 1 - \sum_{j=1}^p \phi_j B^j$; $\Theta(B) = 1 + \sum_{j=1}^q \theta_j B^j$ - полиноми, предполагается, что все корни уравнений $\Phi(z) = 0$, $\Theta(z) = 0$ по модулю больше единицы,

$$(1-B)^d = 1 - dB + \frac{d(d-1)}{2} B^2 - \dots = \sum_{j=0}^{\infty} \psi_j B^j,$$

$$\psi_j = \prod_{0 < k < j} \frac{k-1-d}{k} = \frac{\Gamma(j-d)}{\Gamma(j+1)\Gamma(-d)}, \quad j = 0, 1, 2, \dots$$

ε_t - гауссовский белый шум $WN(0, \sigma^2)$,

$\Gamma(x)$ – гамма-функция, которая определяется формулой

$$\Gamma(x) = \begin{cases} \int_0^{\infty} t^{x-1} e^{-t} dt, & x > 0, \\ x^{-1} \Gamma(1+x), & x < 0, x \neq -1, -2, \dots \end{cases} \quad (3)$$

В частном случае, при $p, q = 0$ и $d \in (-1/2, 1/2)$, процес имеет вид:

$$X_t = (1-B)^{-d} \varepsilon_t = \sum_{j=0}^{\infty} \psi_j \varepsilon_{t-j}, \quad (4)$$

де $\psi_j = \frac{\Gamma(j+d)}{\Gamma(j+1)\Gamma(d)}$.

Оценку параметра d можно получить из равенства $d = H - 0.5$, где H - показатель Херста [6]. Если $H > 0.5$, то он также указывает на наличие стохастического тренда, который не может быть удален дифференцированием. Для вычисления показателя H известный британский гидролог Х.Е. Херст предложил метод нормированного розмаха (R/S - анализ). Кроме того, на основе R/S анализа вычисляется R/S статистика для проверки статистической гипотезы о наличии долгосрочной зависимости. Процедура проверки гипотезы подробно описана в [7]. R/S - анализ является простым процессом, но он требует обработки большого количества данных. Благодаря развитию компьютерных технологий созданы программы, с помощью которых вычисляют коэффициент Херста. Одна из них Fractan 4.4. Эта программа распространяется бесплатно и помещена на многих Internet-сайтах, например, <http://soft.softodrom.ru/ap/Fractan-p44195>.

3. *Анализ и моделирование индекса ПФТС.* Исследование значений индекса ПФТС по годам с 2000 по 2010гг.(данные сайта www.pfts.ua) показали, что в этих рядах присутствует стохастическая составляющая. Об этом свидетельствуют оценки спектральной плотности и корреляционной функции: периодограмма и коррелограмма (Рис.1 и Рис.2).

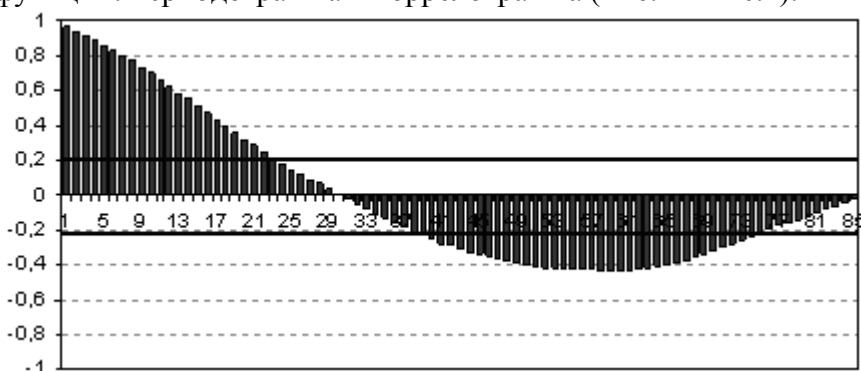


Рис.1. Коррелограмма значений индекса ПФТС за 2010г.

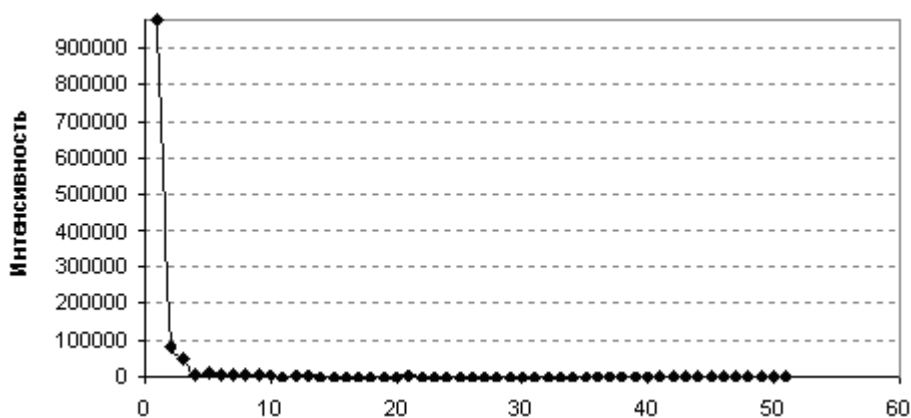


Рис.2. Периодограмма значений индекса ПФТС за 2010г.

На Рис.1 и Рис.2 изображены их графики по данным за июнь-декабрь 2010г. Коррелограмма медленно убывает на лагах 1-20, периодограмма неограниченно возрастает на низких частотах. Анализ проведен с помощью пакета AtteStat. Аналогичный результат получен по данным за все годы рассматриваемого периода. Таким образом, для моделирования и прогнозирования целесообразно использовать дробно-интегрированную модель.

Моделирование в пакете Fractan требуется более трех тысяч эмпирических данных. В настоящее время, благодаря простоте практического применения, особую популярность для анализа и прогнозирования временных рядов приобретают приложения – нейроимитаторы, использующие для анализа и прогноза нейронные сети различных архитектур. Например,

свободно распространяемые - NeuroPro и NeuroShell (<http://www.orc.ru>). NeuroPro 0.25. Пакеты содержат программу, настройку Excel, которая дает возможность применять нейронные сети из рабочих листов Excel. Однако, применение нейроиммитаторов для решения практических задач является «скорее искусством, чем наукой» [8], так как выбор многочисленных параметров, которые требуются для моделирования, осуществляется только на основе личного практического опыта разработчика. Тем не менее, на практике для моделирования предпочитают использование именно нейроиммитаторов.

На Рис.3 показаны результаты моделирования и прогнозирования на 20 дней с помощью нейроиммитатора NeuroPro 0.25. Черным цветом обозначены фактические данные, серым – прогнозные.

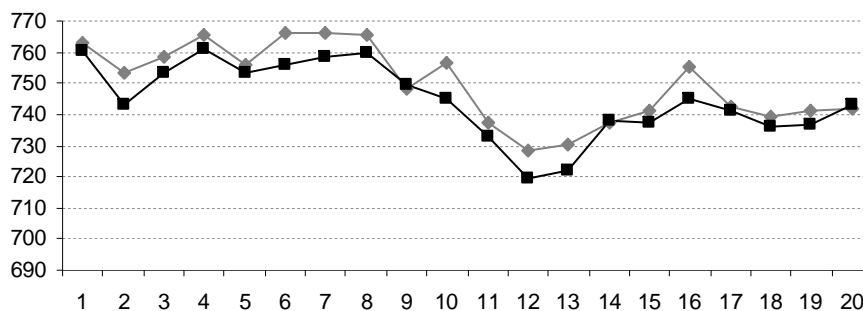


Рис.3. Результат нейромоделирования

Из рисунка видно, что модель полностью отражает тенденцию данных, при этом среднеквадратическая ошибка $\sigma = 6,39$. Таким образом, можно считать модель адекватной, а прогноз вполне удовлетворительным.

Выводы. Для построения адекватной модели, которую можно использовать для описания динамики ряда и прогнозирования его будущих значений, необходимо выяснить природу этого ряда. Наличие стохастического тренда свидетельствует о нестабильности индекса ПФТС и инерционности фондового рынка, что дает возможность под другим углом взглянуть на работу финансовых рынков и методологию оценки рисков.

Литература

1. Wold Н. A Study in the Analysis of Stationary Time Series. /Н. Wold// Сб.научн.трудов. Stockholm: Almqvist and Wiksel, 1938.
2. Канторович Г.Г. Анализ временных рядов/ Г.Г. Канторович// Экономический журнал ВШЭ -№2. – 2002.- С. 252-273.
3. Бокс Дж., Дженкинс Г. Анализ временных рядов. Прогноз и управление/ Дж.Бокс, Дженкинс Г. - Вып. 1, 2 - М.:Мир, 1974.-197с.
4. Granger C.W.J. Some Properties of Time Series Data and Their Use in Econometric Model Specification /C.W.J. Granger C.W.J.// Journal of Econometrics, 1981.-Vol.16.- №1.- P. 121-130.
5. Леоненко М.М. Теоретико-ймовірнісні та статистичні методи в економетриці та фінансовій математиці/М.М.Леоненко, Ю.С.Мішура, В.М. Пархоменко, М.Й. Ядренко - К.: Інформтехніка, 1995. – 380с.
6. Петерс Э. Фрактальный анализ финансовых рисков./Петерс Э. Интернет-трейдинг М.:2004. -304 с.
7. Lo A.W. Long Term Memory in Stock Market Prices/ A.W.Lo //Econometrica.-1991.- №59.-P.1279-1313.
8. Хайкин С. Нейронные сети: полный курс, 2-е изд., испр./ Хайкин С. Пер. с англ. – М.: ООО “И.Д. Вильямс“, 2006. - 1104 с./ Под ред. д.т.н. Н.Н. Куссуль.

УДК 004.272

ПЕРСПЕКТИВЫ ИСПОЛЬЗОВАНИЯ СВОБОДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ СОЗДАНИЯ ТРЕХМЕРНЫХ ИНТЕРАКТИВНЫХ ПРИЛОЖЕНИЙ

Бабенко Е.В. Навка Е.А. Оверчик О.М.

Донецкий национальный технический университет, кафедра компьютерных систем
мониторинга
E-mail: katty_@rambler.ru

Аннотация:

Бабенко Е.В., Навка Е.А., Оверчик О.М. Перспективы использования свободного программного обеспечения для создания трехмерных интерактивных приложений.

Стремительное развитие технологий трёхмерного моделирования и большое количество программных средств, позволяющих выполнять проекты данного направления столь велико, что при выполнении работ в области трехмерного моделирования целесообразным является специальное исследование, позволяющее определить оптимальный набор программных средств. В результате поиска оптимального и рационально использующего техническую базу программного продукта (с точки зрения экономии времени и средств, как технических, так и материальных) нами была выбрана свободно распространяемая система «Blender», с помощью которого можно создавать реалистичные и качественные проекты. В докладе рассматривается использование системы «Blender» для реалистичного моделирования угольной шахты.

Постановка задачи

Задача проекта – разработать многоцелевую трехмерную модель шахты и базу данных объектов, на основе которых может быть создан ряд обучающих видео и приложений. Это позволит людям, заинтересованным в работе, устройстве, просто прогулке в шахте, но не имеющим возможности реального спуска к недрам Земли, узнать больше о скрытой стороне Донбасса. Одно из направлений, ознакомительного приложения, позволит желающим отправиться в увлекательное и не требующее специальных условий путешествие. Проект позволит гостям и жителям нашего региона хотя бы раз в жизни почувствовать себя на месте тех людей, которые сделали возможным строительство и развитие города Донецка. Ведь многие из нас не задумываются о том, какой колоссальный труд - быть шахтёром...

В Донбассе угольная промышленность является одной из основных отраслей. ДонНТУ занимается обучением ведущих специалистов в этой области. Чтобы сделать обучение более эффективным, мы предлагаем разработанный нами проект, который будет помогать студентам осваивать азы техники безопасности, учиться быстро и правильно реагировать на разнообразные непредвиденные ситуации в шахте. Назначение интерактивного приложения – подготовить выпускников нашего вуза к работе в реальной шахте.

Итак, исходя из актуальности для нашего региона, связанной с развитием угольной промышленности, можно с уверенностью сказать, что наша работа, является перспективным началом в данном направлении разработок. Так как основной целью и задачей нашего проекта, является разработка приложений и создание баз данных трехмерных объектов, которые позволят быстро создавать новые, необходимые для работы и обучения модели. Можно с уверенностью сказать, что данная работа имеет перспективы развития и будет пользоваться спросом.

Применение трехмерной графики довольно обширно, начиная с разработки дизайна мелкой бытовой техники, заканчивая моделированием крупных промышленных объектов. В работе над проектом было принято решение использовать свободные программные продукты, не требующие покупки лицензии, позволяющие создавать подобные проекты. Тем самым использование трехмерной модели шахты в нашем проекте позволяет дистанцировать обучаемых горному делу от непосредственного пребывания в опасной ситуации, которая может случиться во время работы и тем самым позволяет будущим шахтерам смоделировать линию поведения при возникновении аварийных ситуаций.

Особенности использования системы «Blender».

Самым главным достоинством «Blender» отличающим его от многих других популярных пакетов таких как «3dsMax», «Maya», является то, что это свободный пакет для создания трехмерной компьютерной графики. При этом он почти не уступает в функциональности коммерческим проектам. Главным недостатком этого программного продукта «Blender» считается то, что он имеет очень неудобный для пользователя интерфейс и с точки зрения начинающего специалиста является трудным в изучении и работе с ним.

Кроме стандартных средств моделирования, анимации, рендеринга, постобработки видео «Blender» также содержит средства для создания интерактивных игр. Если бы для разработки модели были выбраны такие популярные программы «3dsMax» или «Maya», то перед нами возникло сразу несколько проблем:

- Во-первых, для создания интерактивности пришлось бы дополнительно использовать какой-нибудь из уже существующих игровых «движков». Это бы значительно увеличило время выполнения работы, т.к. для начала необходимо ознакомиться с принципами работы этого «движка», научиться с ним работать, а затем загрузить созданные модели и сделать приложение.

- Во-вторых, не всякий игровой «движок» способен правильно экспортировать файлы сцены созданные в «3dsMax», «Maya» и т. д. У большинства из них отсутствует поддержка сложных материалов (Blinn, Anisotropic), Sub-D, и многих часто используемых источников освещения (Area light, Spot light).

При использовании «Blender» таких проблем не возникает. Изначально в «Blender» встроен Game Engine, позволяющий достаточно быстро создавать простые приложения с использованием объектов, созданных в Blender без потери качества.

«Blender» развивается медленнее, чем подобные ему, но коммерческие проекты. Для него позднее появляются специальные обновления и дополнения, в частности: частицы, динамика жидкостей, управление анимацией и т.п. Нет возможности управлять памятью при рендеринге и анимации (в «3ds Max», «Maya» можно выделять необходимое количество оперативной памяти, которое будет использовать этот процесс). Одним из основных достоинств Blender, является то, что этот программный продукт занимает оптимальное количество как оперативной, так и физической памяти, так как в нём в разы меньше функционально сложных структур. Но для разработки и проектирования узкопрофильных приложений (данная работа) это не имеет особого значения, так как для придания более реалистичного вида модели нами не были использованы усложнённые алгоритмы прорисовки эффектов. В случае, когда нам необходимо создать правильно оформленную сцену, Blender позволяет грамотно подобрать текстуры, либо добавить нужное изображение (продолжение коридора), видео (огонь). Разрабатывать, дополнять и обновлять сцены и базы объектов можно с максимальной скоростью и минимальной затратой ресурсов и времени.

Для создания проекта, было рассмотрено несколько подобных работ. В том числе,

для более тщательного и глубокого анализа, нами был изучен проект польской лаборатории методов моделирования и эргономии «Komag». В котором был смоделирован взрыв метана, произошедший шахте каменного угля им. Засядько. Данная реконструкция, в полной мере отображает взрыв, но не даёт возможности рассмотрения внутреннего обустройства шахты с целью ознакомления или более точного моделирования и описания специфики взрыва. Самое существенное отличие нашего проекта от работы, созданной в польской лаборатории «Komag» заключается в том, что данное приложение, может быть использовано в обучении, так как технические особенности его позволяют ознакомиться с шахтой в игровом режиме, а расширенная база материалов и конструкций, позволяет быстро и качественно моделировать различные ситуации.

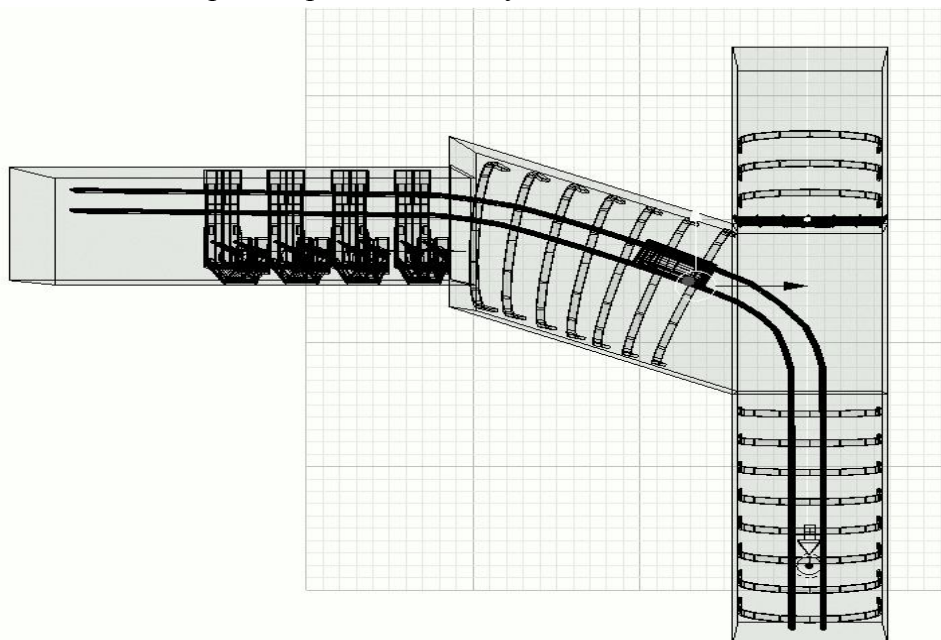


Рис. 1 – Двумерная схема моделируемого участка

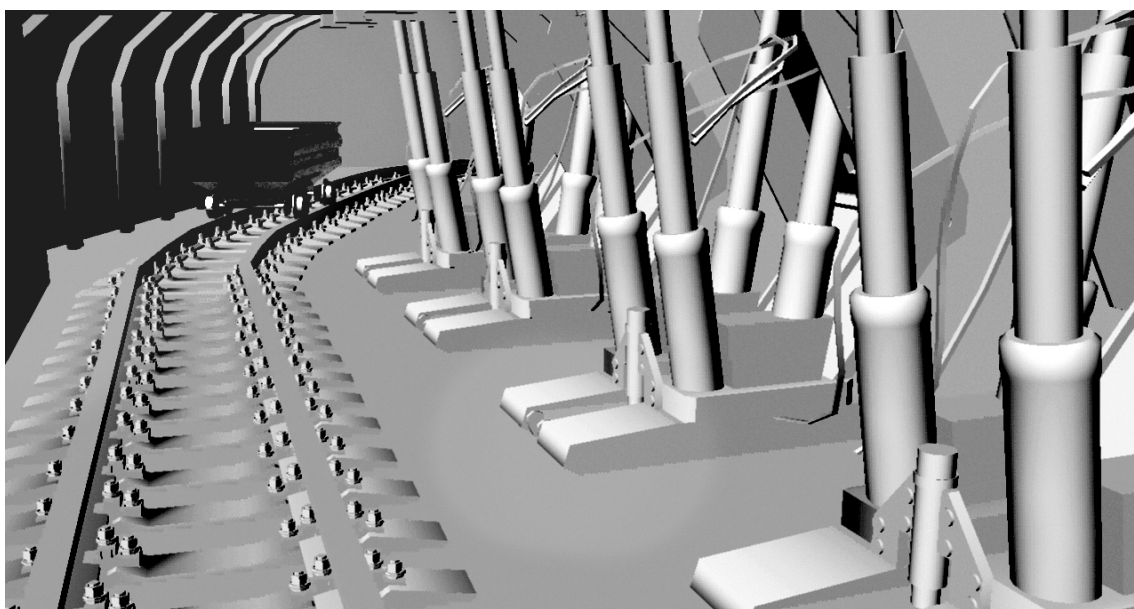


Рис. 2 – Трёхмерная модель участка

Полученные результаты

На данном этапе разработана модель, включающая участок штрека с рельсами, вагонеткой и механизированной крепью (рис. 1 и 2).

Выводы

В дальнейшем, на основе проекта, можно свободно генерировать видеоролики и специальные приложения для различных целей. Динамично развивать направление дистанционного обучения студентов, так как это позволит использовать теорию на практике в реальном времени, просто на обычном занятии. Так же проект может послужить отличной основой, для «виртуальных путешествий» туристов, которые прибудут в наш город на «Евро-2012». Так как наш город не сможет организовать реальные музеи-шахты, в связи с техническими проблемами (невозможность спуска в шахту неподготовленных людей), то данная альтернатива весьма привлекательна.

Развитие угледобывающей промышленности требует более тщательного подхода к изучению и работе в данном направлении. Наша работа является очевидным показателем того, что полезность таких проектов неопределима, как в развитии виртуального туризма так и в обучении молодого поколения шахтёров.

В дальнейшем, нашей целью может стать коммерческое использование проекта в рамках нашего региона.

Литература:

1. Шахти і кар'єри/ интернет - ресурс. – режим доступа: [www/URL: http://center.donetsk.ua/shaxty-i-karyery/donecka-vugilna-shaxta-800-metriv-do-soncyu/](http://center.donetsk.ua/shaxty-i-karyery/donecka-vugilna-shaxta-800-metriv-do-soncyu/)
2. Восстановление промышленного оборудования/ интернет-ресурс. – режим доступа: [www/URL: http://igm.com.ua/](http://igm.com.ua/)
3. Шахтное оборудование/ интернет – ресурс. – режим доступа: [www/URL: http://elektromagnit.com.ua/category/shaxtnoe-oborudovanie](http://elektromagnit.com.ua/category/shaxtnoe-oborudovanie)
4. Blender-Википедия [текст]/ интернет – ресурс. – режим доступа: [www/URL: http://ru.wikipedia.org/wiki/Blender](http://ru.wikipedia.org/wiki/Blender)
5. 3D Modeling, Animation and Rendering Software – Autodesk/ интернет – ресурс. – режим доступа: [www/URL: http://usa.autodesk.com/3ds-max/](http://usa.autodesk.com/3ds-max/)
6. RENDER.RU->уроки->maya/ интернет – ресурс. – режим доступа: [www/URL: http://www.render.ru/books/index.php?book_cat=4](http://www.render.ru/books/index.php?book_cat=4)
7. Гришко, А.П. Стационарные машины. Т. 2. Рудничные водоотливные, вентиляторные и пневматические утановки: Учебник/ А.П. Гришко// miningbook.ru – 2007. 586 с.
8. Попов, В.Н., Букринский, В.А., и др. Геодезия и маркшейдерия: Учебник/ В.Н. Попов, В.А. Букринский и др.//miningbook.ru – 2007. 493 с.

УДК 004.45

ТЕХНОЛОГИИ ВИРТУАЛИЗАЦИИ КАК СРЕДСТВО ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ КОМПЬЮТЕРНОЙ ПОДДЕРЖКИ УЧЕБНОГО ПРОЦЕССА

Власенко А.П., Аноприенко А.Я.

Донецкий национальный технический университет

Кафедра компьютерной инженерии

E-mail: commando.ua@gmail.com

Аннотация

Власенко А.П., Аноприенко А.Я. Технологии виртуализации как средство повышения эффективности компьютерной поддержки учебного процесса. Рассмотрены методы модернизации учебного процесса с помощью использования технологий виртуализации. Рассмотрены наиболее перспективные решения виртуализации. Определен метод внедрения новых компьютерных решений в учебный процесс с использованием технологий виртуализации.

Общая постановка проблемы

Рассматривая вопрос модернизации системы образования, можно прийти к выводу, что практически любой элемент учебного процесса может быть изменен в лучшую сторону: от незначительных изменений в системе оценивания до кардинальных преобразований в самой методике преподавания.

На сегодняшний день компьютерные технологии могут предоставить системе образования следующие технические решения для улучшения учебного процесса:

- организовать систему дистанционного обучения с использованием системы видеовещания и электронного документооборота;
- создать онлайн-порталы для студентов и преподавателей, включающие в себя электронные библиотеки и базы методических материалов;
- внедрить единую компьютеризированную систему оценивания знаний;
- создать для каждого студента уникальные рабочие условия для получения практических навыков.

К сожалению, многие из этих решений до сих пор не нашли должного применения в образовательном процессе. Так, например, внедрение компьютеризированной системы оценивания знаний обычно является попыткой полностью отстранить преподавателя от обработки результатов. В этом и заключается ошибка - изменения должны быть планомерными, фактически, постепенно улучшая условия как преподавания, так и обучения. Главным должно быть создание соответствующих условий и инфраструктуры, на базе которых и будут появляться все новые и новые возможности.

В докладе рассматривается способ применения технологии виртуализации для создания основы модернизированного образовательного процесса. Технология виртуализации может быть использована на всех уровнях системы образования - в учебном процессе и в его организации, а также в работе программной и аппаратной части, на которой и базируются все приведенные выше решения.

Виртуализация и внедрение новых решений в учебный процесс

Технология виртуализации нашла свое применение во многих отраслях, где необходима унификация условий, повышенная защищенность и высокая производительность [1]. При создании основы для системы обучения унификация условий необходима в первую очередь.

Для достижения унификации условий в рамках технологии аппаратной виртуализации аппаратное обеспечение некоторой платформы эмулируется на базе программной или

аппаратной системы. При этом процесс развертывания представляет собой несколько простых шагов [2]:

- 1) установку гипервизора виртуальных машин;
- 2) копирование функционального блока;
- 3) конфигурацию функционального блока для работы с конкретными задачами.

Конкретным примером применения технологии виртуализации может служить развертывание инфраструктуры новообразованного факультета в университете. Основной задачей является создание базовых образов и виртуальных машин, которые выполняют необходимые функции. В идеале, время развертывания может занять менее одного дня:

- 1) в дата-центре университета выделяются и подготавливаются физические ресурсы для размещения сервисов факультета и клиентских приложения для взаимодействия с ними;
- 2) на серверное оборудование копируются виртуальные машины, содержащие в себе сервисы для функционирования инфраструктуры факультета;
- 3) на клиентское оборудование копируются образы машин, содержащие в себе клиентское программное обеспечение для работы с сервисами факультета.

Таким образом, с помощью постепенного развертывания унифицированных виртуальных машин и образов, содержащих необходимые сервисы, новые технологии могут быть внедрены в существующую систему образования. Фактически, с применением технологии виртуализации, развертывание всех компьютеризированных сервисов будет максимально упрощено, вне зависимости от оборудования, на котором будет производиться развертывание.

Применение технологий виртуализации для модернизации видеовещания конференций

Вопрос применения новых решений встал во время организации конференции «ИКТ-2010», проходившей в Донецком национальном техническом университете в ноябре 2010г. Для организации видеовещания секций конференции требовались следующие типы оборудования (рис. 1):

- сервер видеотрансляций;
- серверные машины с видеокамерами для передачи видеопотока на сервер видеотрансляций.

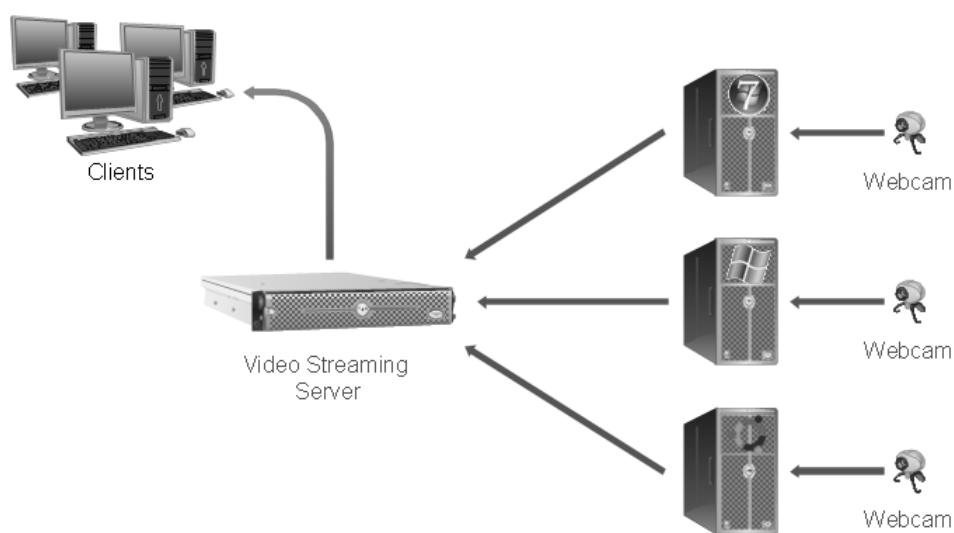


Рис. 1. Схема организации видеовещания конференции «ИКТ-2010»

Сложность организации видеовещания заключалась в том, что каждая машина, оснащенная видеокамерой, базировалась на различных операционных системах и

микропроцессорных архитектурах – Microsoft Windows XP x86, Microsoft Windows 7 x86-64, Linux x86-64. То есть, для обеспечения работы видеовещания требовалась тонкая настройка каждого из компьютеров, что отняло большое количество времени, а в других случаях, например, при отсутствии программного обеспечения под определенную операционную систему и микропроцессорную архитектуру, было бы и вовсе невозможно. Именно в такой ситуации использование образов машин необходимо. Их развертывание заняло бы минимум времени.

Вторая часть проблемы заключалась в том, что сервер видеотрансляции пришлось внедрить в существующую инфраструктуру, тем самым, понизив ее безопасность.

Для оптимизации и модернизации процесса видеовещания предложена схема (рис. 2), которая сокращает время развертывания и исключает возможные проблемы. В данной схеме организации используется два типа оборудования:

– первый тип – с виртуализацией на базе гипервизора виртуальных машин Xen [2] (Video Streaming Server);

– второй тип – образ сервера, работающего с веб-камерами, запускаемый с внешнего носителя.

То есть, на время проведения конференции на серверах развертывается заранее подготовленная среда со всеми заготовленными сервисами.

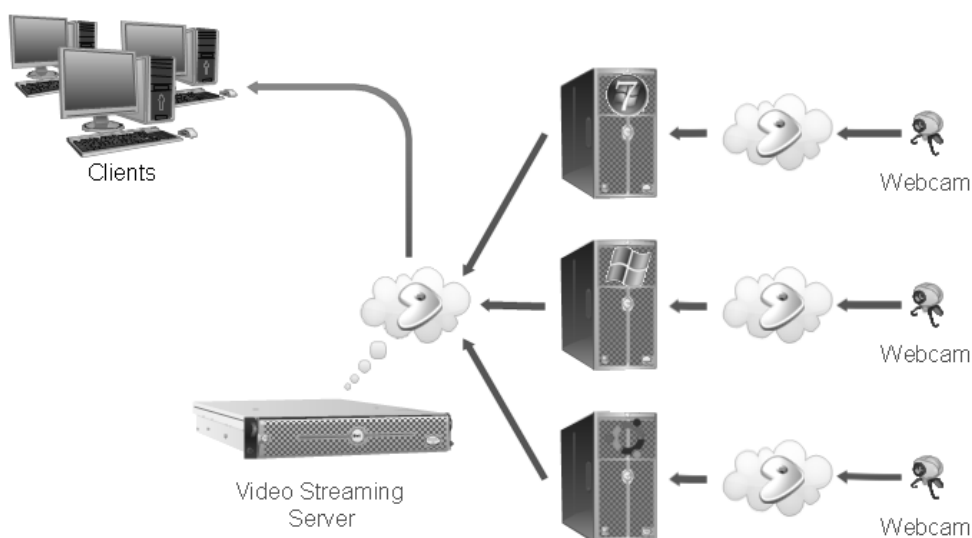


Рис. 2. Модифицированная схема организации видеовещания

Виртуализация с использованием гипервизора Xen

Остановимся подробнее на схеме виртуализации с помощью гипервизора Xen, которая может стать основополагающей для всех будущих проектов. Основными преимуществами использования виртуализации на базе гипервизора Xen являются следующие [3]:

- повышение изоляции между различными сервисами, что снижает вероятность сбоев от взаимного влияния программ и повышает безопасность;
- решение проблем администрирования – возможность ограничить права каждого администратора только самыми необходимыми;
- снижение потенциальных вредных последствий взлома какой-либо из служб;
- распределение ресурсов – каждая машина получает столько ресурсов, сколько ей необходимо, но не более того;
- гибкое распределение сетевого трафика между машинами;

- возможность live-миграции машин и постепенного обновления критических серверов;
- возможность выполнения регрессионных тестов;
- возможность экспериментирования и исследования.

Рассмотрим подробнее способы настройки и использования гипервизора Xen. К числу аппаратных требований относится процессор, поддерживающий технологию виртуализации (следует отметить, что большая часть современных процессоров поддерживает данную функцию). Для функционирования гипервизора Xen должна быть установлена операционная система, работающая в домене 0, – это могут быть операционные системы Linux, NetBSD, OpenSolaris [2]. На установленную операционную систему ставятся пакеты, содержащие утилиты Xen, а также модифицированное ядро. Конфигурируется загрузчик, который загружает новое ядро. После перезагрузки гипервизор Xen готов к работе.

На запущенном гипервизоре создаются конфигурационные файлы для виртуальных машин, а также виртуальные разделы, на которые, как на обычный жесткий диск, ставятся операционные системы, настраиваются нужные сервисы и службы (рис. 3). Для организации видеовещания в виртуальной машине планируется установить операционную систему Gentoo Linux i386 и сервер видеовещания VideoLAN. При необходимости выделения дополнительного сервера видеовещания следует создать копию существующей системы и реконфигурировать ее.

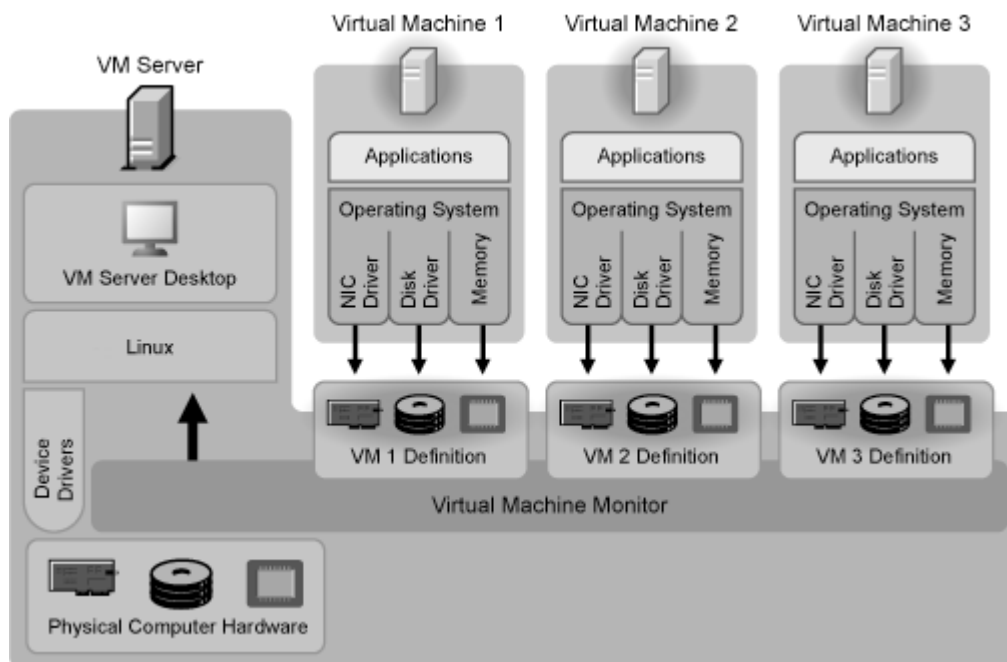


Рис. 3. Монитор виртуальных машин Xen в архитектуре сервера [4]

Для реализации прочих серверов используется аналогичная схема:

- создается виртуальная машина;
- устанавливается подходящая операционная система (Windows, Linux, BSD);
- устанавливаются обслуживающие серверы (WEB-сервер или DNS-сервер, FTP-сервер, сервер конференц-связи) или любые другие необходимые службы.

Обновляемая система служит основой для других систем. Машину, содержащую в себе WEB-сервер, можно дублировать для каждого сайта, при этом, будет обеспечена максимальная защита и изоляция этого WEB-сервера и сайта от уязвимостей других сайтов.

Для эффективной организации процесса обновления программного обеспечения можно использовать машину-репозиторий. Рекомендуется использовать единый дистрибутив

операционной системы GNU/Linux, например Gentoo Linux или Ubuntu Linux. При использовании операционной системы Gentoo Linux к задачам репозитория относятся компиляция новых пакетов из исходных кодов и последующая раздача обновлений в виде бинарных пакетов для всех остальных виртуальных машин. Таким образом, достигается экономия в трафике и процессорном времени, т.к. каждой машине не понадобится компилировать обновления по отдельности. Обновления в операционной системе Ubuntu Linux не требуют компиляции, поэтому экономия будет получена лишь в интернет-трафике.

В настоящее время набирают популярность VDS и VPS хостинги, базирующиеся на технологии Xen. В отличие от аренды физического выделенного сервера, аренда виртуального выделенного сервера позволяет четко запросить необходимое количество физических ресурсов, при этом, не переплачивая за неиспользуемые. Управление операционной системой внутри виртуального сервера практически ничем не отличается от управления операционной системой на физическом сервере. Клиенты VDS\VPS хостинга имеют root-доступ, собственные таблицы маршрутизации и IP-адреса. Обычно, стоимость VPS\VDS хостинга занимает промежуточное место между стоимостью виртуального хостинга и арендой выделенного сервера.

Виртуализация с использованием гипервизора виртуальных машин Xen внедряется на кафедре компьютерной инженерии Донецкого национального технического университета с марта 2011 года. Первый сервер, который построен по данной технологии, будет содержать образы для WEB-сайтов кафедр факультета компьютерных наук и технологий и другие сервисы.

Выводы

Гипервизор Xen зарекомендовал себя в корпоративном секторе и может найти свое применение в системе образования. Внедрение технологий виртуализации на базе гипервизора Xen позволит выделить для каждого студента свой собственный «виртуальный» компьютер для экспериментов, с набором необходимых программных продуктов для обучения и научных исследований, а также с правами администратора. При этом повышается уровень защиты как серверов университета, так и личного пространства каждого студента. Запускаемые приложения не могут навредить серверу факультета или внедриться в системы других студентов. К личному пространству студента будет иметь доступ лишь он сам – возможность взлома прав доступа к файлам будет исключена, т.к. каждый студент работает на собственной изолированной машине.

Таким образом, технологии виртуализации являются мощным инструментом модернизации компьютерной поддержки системы обучения. Они позволяют модульно и постепенно внедрять унифицированные новые технологии в систему образования, при этом рационально используя физические ресурсы и обеспечивая защищенность информации.

Литература

1. Virtual Linux : An overview of virtualization methods, architectures, and implementations [Electronic resource] / M. Tim Jones / Интернет-ресурс. - Режим доступа : [www/ URL: http://www.ibm.com/developerworks/linux/library/l-linuxvirt/](http://www.ibm.com/developerworks/linux/library/l-linuxvirt/)
2. Running Xen: A Hands-On Guide to the Art of Virtualization [Текст] / Jeanna N. Matthews, Eli M. Dow, Todd Dshane, Wenjin Hu, Jeremy Bongio, Patrick F. Wilbur, and Brendan Johnson // Prentice Hall, 2008.
3. Преимущества виртуализации [Electronic resource] / Интернет-ресурс - Режим доступа: [www/ URL: http://xgu.ru/wiki/Преимущества_виртуализации](http://xgu.ru/wiki/Преимущества_виртуализации)
4. An Introduction to Virtualization [Electronic resource] / Интернет-ресурс – Режим доступа: [www/ URL: http://tr.opensuse.org/An_Introduction_to_Virtualization](http://tr.opensuse.org/An_Introduction_to_Virtualization)

УДК 004

Использование Rational Rhapsody для эффективной разработки и тестирования моделирующих систем на базе UML**Губский А.Е. Андрюхин А.И.**

Донецкий Национальный Технический университет
кафедра прикладной математики и информатики
E-mail: oleksandr.gubsky@gmail.com

Аннотация:

Губский А.Е. Андрюхин А.И. Использование Rational Rhapsody для эффективной разработки и тестирования моделирующих систем на базе UML. Рассмотрены основные возможности среда разработки на основе визуального проектирования для разработчиков встраиваемых систем и программного обеспечения реального времени - IBM Rational Rhapsody.

Общая постановка проблемы

Для соответствия рыночным требованиям и сохранения лидирующих позиций важно использовать самые современные инструменты разработки. За последнее десятилетие основой таких инструментов стал язык UML, признанный стандартом де-факто в создании сложных систем и программного обеспечения. Безусловно, самым эффективным инструментом, реализующим его возможности с учетом особенностей встраиваемых систем, является среда разработки IBM Rational Rhapsody.

Исследования

IBM Rational Rhapsody – среда разработки на основе визуального моделирования с использованием языка UML (Unified Modeling Language). Rhapsody пересматривает традиционный процесс разработки встраиваемых систем и приложений реального времени, устраняя традиционные барьеры между этапами процесса разработки основанного на документации.

Rhapsody предоставляет системным инженерам, разработчикам ПО и тестировщикам общую среду разработки на основе визуального моделирования, в которой можно проанализировать требования, спроектировать систему и ПО, сгенерировать и разработать приложение, а также быстро, эффективно и своевременно протестировать текущие результаты на любом этапе процесса разработки: от анализа требований до готовой встраиваемой системы.

Model Driven Architecture ® (MDA ®) - подход к использованию модели разработки программного обеспечения, включая написание спецификаций и фактической разработки приложений. Этот подход позволяет отделить от деталей реализации функциональные возможности и поведение системы. Это обеспечивает независимость приложения от среды. Сначала создается одна или более независимых от платформы модели (PIM), которые затем переводятся на один или более конкретные платформы моделей (PSM). MDA - включает широкий круг понятий, которые применяются ко всем проектам разработки программного обеспечения в различных направлениях, как электронная коммерция, финансовые услуги, здравоохранение, аэрокосмическая и транспортная отрасли. Разработчики встраиваемых систем должны сосредоточить внимание на их конкретные потребности в целях обеспечения максимальной выгоды MDA при выполнении их особых потребностей в реальном времени: компактный код, безопасность, надежность и управление специализированным оборудованием.

Rhapsody является лидирующей на рынке UML 2.1 Model Driven Development (MDD) решений, поскольку он использует ряд перспективных технологий, которые обеспечивают пользователям наиболее эффективные средства производства систем и программного обеспечения. Rhapsody всегда сосредоточен на потребности разработчиков и полностью поддерживает концепции МДА. Поддержка UML 2.0 в Rhapsody не имеет себе равных и однозначно позволяет преодолевать функциональные и объектно-ориентированные разрывы в одной среде, что дает возможность очень гибко проектировать. Для действительно эффективной разработки на основе моделей, Rhapsody имеет среду разработки, которая держит Вас в постоянной связи с поведением системы посредством выполнения и проверки.

Вы можете использовать IBM Rational Rhapsody с IBM Rational Test RealTime для тестирования компонентов и выполнения анализа. Благодаря интеграции с Rational Rhapsody и Rhapsody IBM Rational TestConductor Add On , вы можете использовать Rational Test RealTime для автоматизации моделирования UML и обеспечивают генерацию кода для поддержки на основе моделей парадигмы model-driven. Rational Test RealTime расширяет разработки на основе моделей будет включать методы тестирования разработчиками, а разработчики могут использовать его для вызова среды выполнения анализа кода, который создается на основе моделей UML.

Графическая среда дает вам понять, как система ведет себя во время выполнения.

Это позволяет свободно работать в рамках исходных файлов, и обеспечивает изменения на уровне исходного кода, в динамически обновляющийся модели, поэтому модель и код всегда синхронизированы.

Подключение с помощью инфраструктуры CM, дает возможность малым и большим коллективам эффективно совместно работать над проектами, будь то команда местных или распространенных по всему миру разработчиков, а также предоставление разного уровня передовых моделей и объединения возможностей.

С помощью Rational Rose ®, IBM Rational ® стал ключевым игроком на рынке систем UML моделирования. Rose имеет архитектурную основу, которая позволяет интегрировать с другими инструментами жизненного цикла разработки для создания полного решения от концепции к целевой системе.

Rose охватывает большую часть процесса разработки, внедрения и тестирования. Цель Rose - уменьшить сложность проектирования систем и программных архитектур и совершенствования сотрудничества за счет использования UML в качестве языка общения, который мог быть легко понятен инженерам. Хотя этот подход пользуется большим успехом, технологическое развитие и усложнение требуют, чтобы пробелы, которые существуют в CASE технологии, имели решение. Model-Driven Development (MDD) является технологией, которая помогает проводить тестирование и внедрение, позволяя вам с легкостью воспользоваться существующей моделью Rose или даже исходным кодом.

Разработки на основе моделей используется UML 2 модели описывают не только дизайн, но и требования и реализации. Технология, на которой построена Rose в основном используется для документирования архитектуры системы. Но нет никакого способа проверить правильно ли построены спецификации и как система будет вести себя, так как документ является статическим. Конечно, если ошибка вводится в системе в ранней стадии разработки, она не может быть обнаружена до полной реализации. На рисунке 1 показано, как дорого могут стоить, ошибки в начале жизненного цикла.

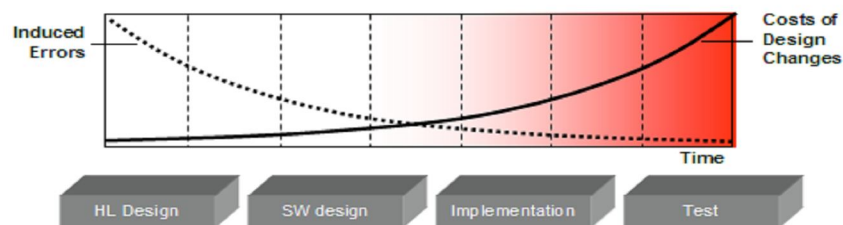


Рисунок 1 — Цена ошибок на этапе моделирования

Цель подхода MDD с Telelogic Rhapsody®, - обнаружить дефекты в начале разработки, поскольку стоимость внесения изменений в этот момент минимальна. Устранение ошибок в дизайне снижает не только стоимость, но и время, проведенное в процессе разработки.

Следующие данные сравнивают результаты использования Rose и Rhapsody.

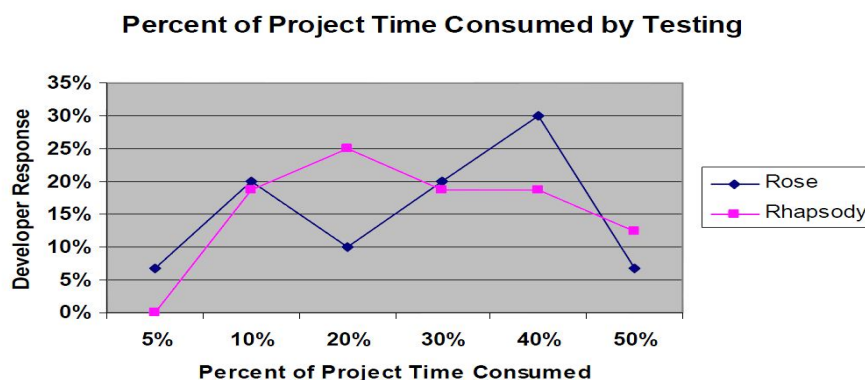


Рисунок 2 — Сравнение результатов

На рисунке 2, мы видим, что Rhapsody пользователи обычно тратят от 10 до 30 процентов своего времени на стадии тестирования, в то время как для Rose, разработчики тратят около 40 процентов своего времени на тестирование. Хотя оба продукта являются успешными в разработке систем, технологий следующего поколения, Rhapsody предлагает дополнительные преимущества: более высокую производительность системы и функциональные возможности, более современной график проекта.

Выводы

На самом базовом уровне, Rose является подмножеством Rhapsody, так как Rhapsody он не только поддерживает новейшие UML стандарты, но также поддерживает стандартные, предметно-ориентированные профили, таких как Systems Modeling Language (SysML), отражено на рис.3.

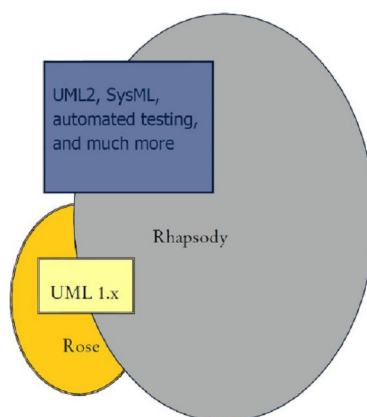


Рис.3.Соотношение Rose и Rhapsody.

Такая гибкость позволяет разработчикам использовать язык дизайна наиболее подходящий для описания своей предметной области. Rhapsody, разработки на основе моделей

платформы, используется в основном для аспектов окружения, которые выходят за рамки простого моделирования. Процесс миграции позволяет пользователям Rational Rose, воспользоваться расширенными возможностями Rhapsody и перенести свои данные в Rhapsody. В настоящее время выполняется разработка нескольких типов моделирующих систем, частности событийного типа.

Литература

1. Rational Rhapsody - <http://www-01.ibm.com/software/awdtools/rhapsody/>
2. Rhapsody. - <http://en.wikipedia.org/wiki/Rhapsody>
3. Основы UML и IBM Rational Rhapsody для разработчиков программного обеспечения
<http://www.swd.ru/index.php3?pid=158>

УДК 004.4+004.94

ВИРТУАЛЬНЫЙ ЛАБОРАТОРНЫЙ ПРАКТИКУМ ПО ФИЗИКЕ**Жуковский Г.Б.**

Донецкий национальный технический университет

Кафедра компьютерных наук и технологий

E-mail: tuborg_first@mail.ru**Аннотация**

Жуковский Г.Б. Виртуальный лабораторный практикум по физике. Рассмотрены этапы создания проекта программной системы дистанционного учебного курса для проведения практических работ и тестирования по курсу общей физики. Выполнен системный анализ виртуального лабораторного практикума. Разработана схема компьютерной обучающей системы. Проанализированы информационные потоки, связывающие практикум с другими информационными системами. Обоснован выбор программных средств реализации практикума по результатам анализа современных инструментальных средств разработки.

Общая постановка проблемы

В современном мире все больше задач можно решить с помощью Интернета. В данной статье рассматривается внедрение виртуальной лаборатории по физике во всемирную сеть. Виртуальная лаборатория предоставляет педагогу и студенту возможность находить наиболее интересные и эффективные методы обучения, делая занятия интересными и более насыщенными.

Виртуальные лаборатории - это новое уникальное направление в системе образования. Ни для кого не секрет, что 90% информация поступает к нам в мозг через зрительный нерв. И не удивительно, что пока человек сам не увидит, он не сможет четко уяснить природу тех или иных физических явлений. Поэтому процесс обучения обязательно должен подкрепляться наглядными материалами. И просто замечательно, когда можно не только увидеть статичную картинку изображающую какое-либо физическое явление, но и посмотреть на это явление в движении. Данные приложения позволяет в легкой и непринужденной форме показать не только действия основных законов физики, но провести он-лайн лабораторные работы по физике по некоторым разделам общеобразовательной программы [1].

Виртуальные лаборатории существуют уже давно и для ДонНТУ это не новшество [2, 3]. Однако возможность использовать такие лаборатории через Интернет для ДонНТУ является нововведением. Система MOODLE поддерживает размещение всевозможных тестов и заданий по разным дисциплинам, но виртуальный практикум по физике в системе дистанционного обучения будет использован впервые.

Системный анализ и разработка общей схемы компьютерной обучающей системы

Виртуальный лабораторный практикум представляет из себя набор из лабораторных работ, охватывающих следующие разделы физики:

- механика;
- молекулярная физика;
- электростатика и постоянный ток;
- электромагнетизм.

Хотя лабораторные работы и принадлежат разным разделам физики, но в них есть и много общего. Например, во многих лабораторных для расчетов используется одно и то же оборудование. Вне зависимости от целей работ, весы всегда взвешивают некие объекты и дают представление об их весе, поэтому нет смысла реализовывать данный прибор каждый раз повторно.

Рассмотрев все возможные варианты, были выбраны такие лабораторные, которые наиболее полно отображают суть законов соответствующих им разделов физики, а также наиболее интересны в проведении экспериментов. Вот их список:

1. Изучение вращательного движения на маятнике Обербека;
2. Определение вязкости жидкости методом падающего шарика;
3. Изучение электростатического поля;
4. Исследование зависимости электрического сопротивления металлов от температуры;
5. Измерение температуры Кюри ферромагнетика.

В целом, виртуальный практикум состоит не только из перечисленных выше лабораторных. Он включает в себя, так же, и общие методические указания по использованию практикума. Эти указания содержат все требования для запуска приложений на компьютере, а так же помощь по использованию каждой конкретной работы. Поэтому структурно-функциональная схема компьютерной обучающей системы имеет следующий вид (рис. 1).

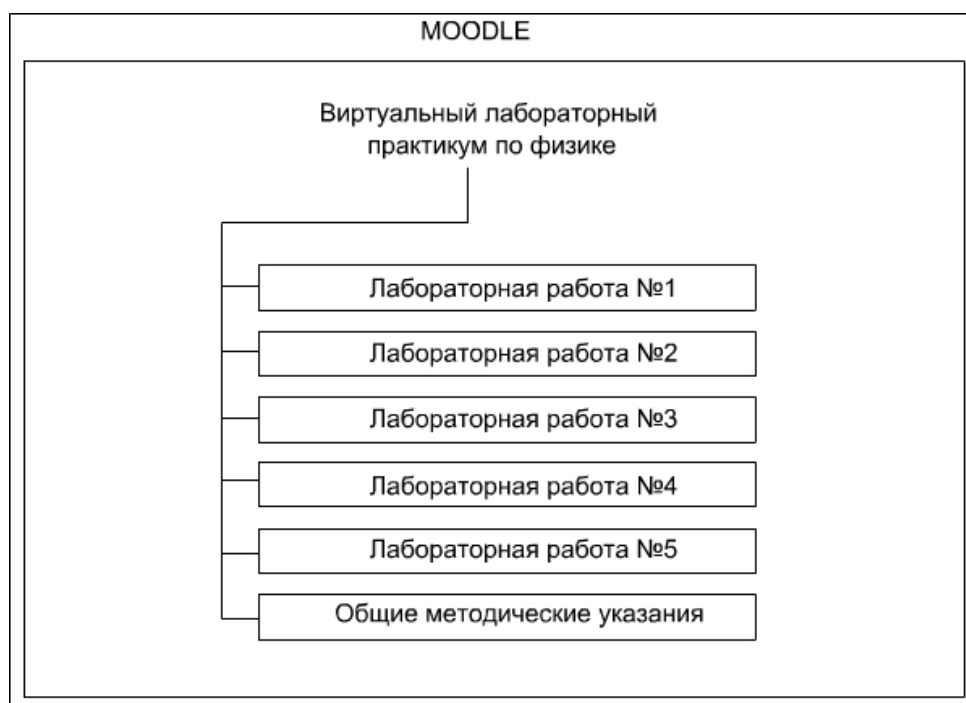


Рис. 1. Структурно-функциональная схема компьютерной обучающей системы

Виртуальный лабораторный практикум доступен через дистанционный курс физики <http://ipo.donntu.edu.ua/dist/index.html>.

Общую схему компьютерной обучающей системы можно увидеть на рис. 2.

Анализ информационных потоков

Потоки данных являются абстракциями, используемыми для моделирования передачи информации (или физических компонент) из одной части системы в другую. Потоки на диаграммах изображаются именованными стрелками, ориентация которых

указывает направление движения информации [4]. Для организации взаимодействия между компонентами предусмотрен модуль "Оглавление", который содержит ссылки на все лабораторные работы и общие методические указания.



Рис. 2. Общая схема компьютерной обучающей системы

Компьютерная обучающая система связана с MOODLE. MOODLE - это программный продукт, позволяющий создавать курсы и сайты, размещаемые в сети Интернет.

Сам MOODLE размещен на главном сервере dist.donntu.edu.ua системы дистанционного обучения ДонНТУ. В целях безопасности существует еще один сервер - tempus.donntu.edu.ua. На этом сервере и размещен виртуальный лабораторный практикум. На рис. 3 приведена подробная схема связей практикума с другими информационными системами.

Стрелки означают следующее:

- 1 - браузер запрашивает с сервера dist.donntu.edu.ua страницу с оглавлением;
- 2 - сервер приказывает MOODLE сформировать страницу с оглавлением;
- 3 - MOODLE запрашивает перечень ссылок на ресурсы с базы данных;
- 4 - база данных возвращает перечень ссылок;
- 5 - MOODLE формирует страницу с оглавлением и сообщает серверу о завершении работы;
- 6 - сервер dist.donntu.edu.ua возвращает браузеру результат работы MOODLE;
- 7 - браузер отправляет URL запрос на сервер tempus.donntu.edu.ua;
- 8 - сервер запрашивает с базы данных ресурс, указанный в запросе;
- 9 - база данных возвращает запрошенный ресурс;
- 10 - сервер tempus.donntu.edu.ua возвращает браузеру запрошенный ресурс.

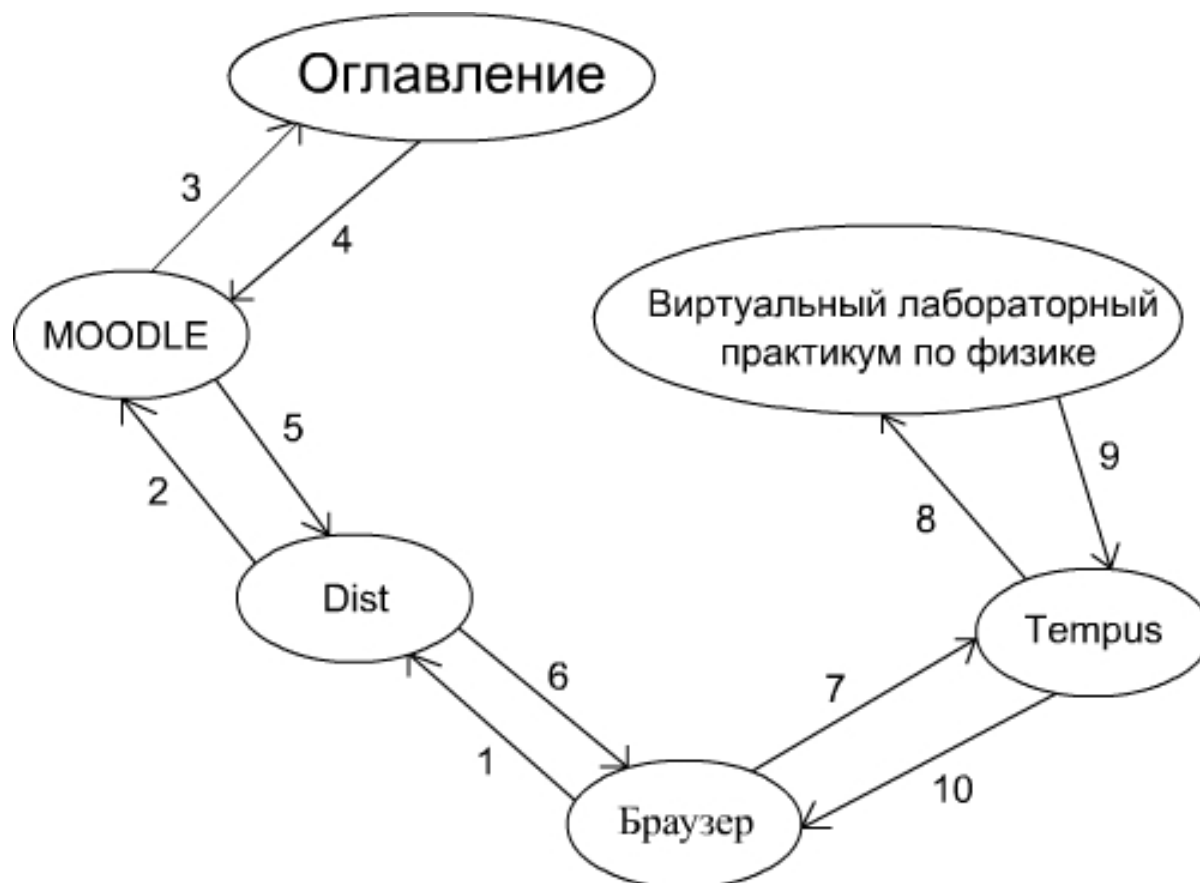


Рис. 3. Схема связей с другими информационными потоками

Стоит заметить, что данная схема не рассматривает случаи отсутствия в базе данных запрошенных браузером ресурсов.

Выбор программных средств реализации

Существует большое количество средств создания виртуальных лабораторных практикумов. Был выделен набор характеристик таких инструментальных средств. Наиболее популярные из них были сравнены по следующим параметрам:

- наличие или отсутствие инструментов эмуляции физических приборов;
- наличие или отсутствие собственного графического редактора;
- требование к установленной виртуальной машине на компьютере пользователя;
- относительная распространенность данной виртуальной машины;
- возможность или невозможность просмотра приложений на мобильных устройствах.

Результаты сравнения предоставлены в таблице 1.

По результатам данной таблицы можно сделать выводы, что наилучшим выбором для выполнения поставленной задачи будет Adobe Flash. Единственный недостаток данной платформы – это отсутствие встроенных инструментов эмуляции физических приборов. Данная проблема не столь существенна, так как Flash обладает мощным графическим редактором и возможностью написания библиотек. С их помощью можно создать совершенно любое виртуальное оборудование. Несомненным плюсом является наивысший уровень распространенности виртуальных машин среди рядовых пользователей Интернета.

Таблица 1 - Сравнение средств реализации виртуальных лабораторных практикумов

	LabView	BARSIC	Flash	Silverlight	JavaFX
Инструменты эмуляции физических приборов	Есть	Есть	Нет	Нет	Нет
Собственный графический редактор	Нет	Нет	Есть	Есть	Нет
Невозможность работы без специального плагина для браузера	Да	Да	Да	Да	Да
Распространенность плагина	Низкая	Очень низкая	Очень высокая	Средняя	Высокая
Возможность запуска приложений на мобильных устройствах	Нет	Нет	Да	Да	Нет

Выводы

В ходе выполнения работы были рассмотрены этапы создания проекта программной системы дистанционного учебного курса для проведения практических работ и тестирования по курсу общей физики. Был выполнен системный анализ виртуального лабораторного практикума. Так же была разработана схема компьютерной обучающей системы. Проанализированы информационные потоки, связывающие практикум с другими информационными системами. По результатам анализа современных инструментальных средств разработки виртуальных лабораторий лучшим был выбран Adobe Flash.

Литература

1. Наглядная физика [электронный ресурс] // Виртуальная образовательная лаборатория. - метод доступа URL: <http://www.virtulab.net/index.php>
2. Дацун Н.Н., Смешков А.А., Товстуха В.С. Построение диаграммы состояния воды в виртуальной лабораторной работе по физике // Образование и виртуальность - 2004. Сборник научных трудов 8-й Международной конфедерации Украинской ассоциации дистанционного образования. - Харьков-Ялта: УАДО, 2004. - с.113-117.
3. Дацун К.А., Товстуха В.С. Виртуальное определение числа Авогардо // Образование и виртуальность - 2003. Под общ. ред. В.А. Гребенюка и В.В. Семенца. - Харьков-Ялта: УАДО, ХНУРЭ, 2003. - с.149-151.
4. Методологии моделирования предметной области [электронный ресурс] // Интернет университет. - метод доступа URL: <http://www.intuit.ru/department/se/devis/6/4.html>

УДК 004

ИССЛЕДОВАНИЕ И СРАВНЕНИЕ ВОЗМОЖНОСТЕЙ NoSQL СУБД Cassandra и РЕЛЯЦИОННОЙ СУБД PostgreSQL В ЗАДАЧЕ ПОИСКА DICOM ИЗОБРАЖЕНИЙ

Карпунов Г.А., Костюкова Н.С.

Донецкий Национальный Технический университет
кафедра прикладной математики и информатики
E-mail: sweet_misery@list.ru

Аннотация

Карпунов Г.А., Костюкова Н.С. Исследование и сравнение возможностей NoSQL СУБД Cassandra и реляционной СУБД PostgreSQL в задаче поиска DICOM изображений. Рассмотрен подход хранения изображений в СУБД Cassandra и СУБД PostgreSQL. Определены преимущества и недостатки схемы данных для хранения PACS DICOM изображений в указанных СУБД. Рассмотрены варианты быстрой работы при разных схемах архитектуры NoSQL СУБД Cassandra.

Общая постановка проблемы

Большинство современных медицинских учреждений имеют MRT сканеры, которые получают данные в цифровом формате, преобразуя их в DICOM файлы. За несколько лет работы MRT сканера накапливаются DICOM файлы исследований объемами в десятки терабайт.

В реляционных базах данных при хранении атрибутов DICOM файлов используется схема «ключ-значение»: каждый атрибут имеет уникальный ключ, по ключу совместно с идентификатором DICOM файла можно получить значение атрибута. Для выборки и отсеивания по нескольким атрибутам в реляционных базах данных необходимо применять стандартную реляционную операцию JOIN при каждом новом атрибуте. Стандарт DICOM подразумевает более 1500 атрибутов, больше половины которых может использоваться в качестве метаданных DICOM файлов. Соответственно, для каждого DICOM файла мы можем получить 1000 атрибутов в базе данных, и вынуждены использовать для отсеивания по набору, например, из 100 атрибутов, 100 операций JOIN. Это приводит к тому, что каждый запрос к небольшому PACS архиву выполняется более 20 секунд, что неприемлемо к системе с потенциально большой нагрузкой.

Необходимо найти альтернативу реляционным базам данным, которая существенно ускорит поиск таких файлов по их метаданным.

Исследования

Для исследования воспользуемся реляционной СУБД PostgreSQL и нереляционной СУБД Cassandra. Объектом исследования будет служить PACS архив разнородных изображений, которые имеют разный размер и, соответственно, разное число атрибутов. Ряд изображений тестовой выборки представляются действительными снимками MRT сканера (5.7 Гб изображений, 46458 DICOM файлов).

Каждый DICOM файл может содержать различное число атрибутов метаданных. В исследовании будут участвовать все метаданные, которые хранятся в DICOM файле. В СУБД PostgreSQL будет использоваться операция JOIN. В системах NoSQL данные хранятся не по реляционной модели. Это подразумевает возможную избыточность данных. Тем не менее, за счет избыточности NoSQL СУБД работают в определенных классах задач значительно быстрее классических реляционных СУБД.

Для эксперимента с СУБД Cassandra будем использовать следующую архитектуру сети (рис. 1):

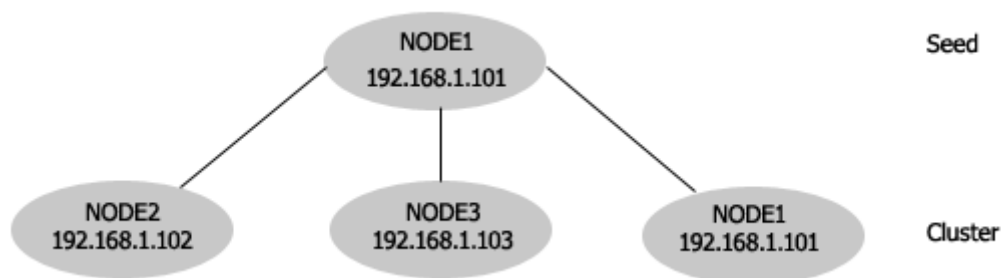


Рисунок 1 – Структура вычислительной сети для исследования

Узлы содержат следующие характеристики (табл. 1):

Таблица 1 – Характеристики узлов сети для исследования

Узел	Процессор	Оперативная память	Скорость HDD
NODE1	Core 2 QUAD Q6600 2.4 ГГц	4096 Мб	7500 об/мин
NODE2	Intel Pentium Dual Core 1.3 ГГц	4096 Мб	7500 об/мин
NODE3	Intel Pentium 3 1 ГГц	256 Мб	7500 об/мин

Сеть между узлами поддерживается с использованием маршрутизатора Cisco Linksys WAG120N, настроенного на скорость передачи данных 100 Мбит/с.

Можно обнаружить, что узел, который распространяет информацию о кластере (NODE1) включен в кластер и работает с данными аналогично NODE2 и NODE3. Получаем кластер, состоящий из 3 узлов (рис. 2).

```

[default@unknown] describe cluster;
Cluster Information:
  Snitch: org.apache.cassandra.locator.SimpleSnitch
  Partitioner: org.apache.cassandra.dht.RandomPartitioner
  Schema versions:
    020b6e96-4f01-11e0-b5a1-bd9004a36f78: [192.168.1.103, 192.168.1.102, 192.168.1.101]
  
```

Рисунок 2 – Информация о кластере

Вследствие того, что реляционные базы данных являются трудно масштабируемыми по горизонтали, сложно масштабировать на имеющуюся сеть СУБД PostgreSQL как минимум с сохранением производительности, которая, наиболее вероятно, в рассматриваемом примере будет ниже. Для эксперимента с СУБД PostgreSQL используется только NODE1 – самый мощный из представленных узлов.

Для добавления данных в базы данных PostgreSQL и Cassandra по выборке из 46458 DICOM файлов сгенерируем файлы с запросом. Для этого извлекаем все атрибуты из каждого файла выборки. Получаем более 47 млн пар вида «ключ-значение».

После занесения данных в схему СУБД PostgreSQL объем базы данных составил 72,65Мб. Данные заносились частями по транзакциям, суммарное время занесения составило 124 минуты. Схема данных представлена на рисунке 3:

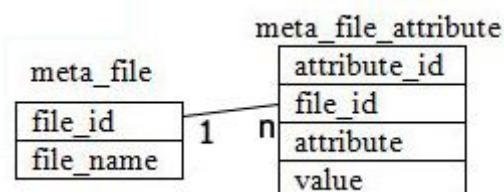


Рисунок 3 – Схема данных для исследования в PostgreSQL

При занесении данных в схему СУБД Cassandra используется внешний файл, загружаемый с помощью утилиты `cassandra_cli`. При загрузке схемы на каждой машине, входящей в кластер, формируется лог транзакции. Большинство NoSQL систем, в отличие от реляционных баз данных, организуют хранилище на основе хеш-таблиц. При этом данные прозрачно масштабируются на множество узлов, входящих в кластер. Вследствие этих условий поддерживать внутренний механизм транзакций в NoSQL становится сложно. При добавлении данных формируется внешний файл транзакции, который сохраняет состояние выполнения большого запроса. Этот файл по мере выполнения запроса обновляется на всех узлах, входящих в кластер. Каждый узел получает только ту часть добавляемых данных с транзакции, которая будет храниться у него в файловой системе. Таким образом, происходит распределение данных между узлами еще в процессе занесения записей.

Алгоритм записи в схему Cassandra:

- создание пространства ключей `dicom`;
- загрузка данных в формате `dicom['file_name']['attribute'] = 'value'`

Запись заняла 49 минут, что примерно в 2 раза быстрее, чем при записи с использованием PostgreSQL. При этом суммарный объем данных, который заняла база данных, на трех узлах составил 93,76Мб. Стоит отметить, что на машине, которая выполняла запрос, сохранилось 79,02Мб данных. Можно обнаружить, что на двух других узлах объем данных выходит одинаковый - 7,36Мб.

Рассмотрим варианты таких отличий в хранении данных. В реляционных базах данных широко применяется резервное копирование. В большинстве случаев резервное копирование делается только несколько раз в неделю. Если отказывает накопитель, может потеряться часть информации.

В NoSQL системах на данном этапе их развития для важных данных необходимо иметь значительную репликацию. При частом обращении к базе данных множества пользователей возникают сложные транзакции, которые иногда могут приводить к потерям данных даже в штатном режиме. В крупных компаниях, которые используют NoSQL системы (Google, Facebook и др.) создаются алгоритмы, которые, при частичной репликации на каждый узел, впоследствии могут собирать данные с разных узлов по одному и тому же ключу, анализировать их корректность и делать вывод об их достоверности.

При запуске Cassandra с узла, который не распространяет информацию о кластере, ему можно указывать параметр `auto_bootstrap`. С установленным параметром узел автоматически скопирует часть данных к себе с уже запущенных узлов. Это также способствует репликации: при отключении одного из узлов, данные, которые он содержал, можно восстановить из реплик, которые содержатся на других узлах. Тем не менее, при отключении значительного числа узлов часть данных может оказаться временно недоступной.

Еще одним недостатком реляционных баз данных является единственная точка входа. В небольших системах нет необходимости в другой архитектуре, потому что такая схема удобна для понимания и управления. Таким образом, если узел, на котором используется база данных, отказывает – отказывает вся система.

NoSQL СУБД предусматривают лучшую отказоустойчивость, так как существует несколько узлов, с которых можно получить данные. В приведенной схеме такой точкой является `NODE1`. В более сложных системах принято 5-30% узлов в кластере запускать в качестве машин, распространяющих информацию о кластере. Вероятность отказа такого числа узлов значительно ниже, соответственно система с такой архитектурой является более надежной.

После занесения данных в базы данных можно исследовать время выполнения запросов в каждой из них. Результаты выполнения различных запросов представлены в таблице 2.

Таблица 2 – Результаты выполнения запросов

Цель запроса	Структура выполнения		Время выполнения	
	Cassandra	PostgreSQL	Cassandra	PostgreSQL
Получить все атрибуты файла	Задание ключом имени файла, результат – все атрибуты	Использование JOIN для добавления каждого атрибута	215 мс	>10 мин
Получить файлы по 4 атрибутам	Задать параметр отсеивания	Использование JOIN для добавления каждого атрибута, четыре JOIN	220 мс	31сек
Получить все файлы, у которых заданный атрибут равен заданному значению	Задаем индекс по атрибуту, ищем по значению	Выбрать атрибут для каждого файла, один JOIN	510 мс	70 сек

Как видно из результатов, полученных при выполнении запросов, скорость NoSQL СУБД превосходит реляционную СУБД PostgreSQL в несколько раз.

В запросе получения всех атрибутов файла очевидно, что при использовании SQL придется использовать JOIN столько раз, сколько файл имеет атрибутов. Таким образом, результирующий запрос будет занимать несколько Кб и выполняться неприемлемо долго. В Cassandra при записи данных задавался ключ – имя файла, поэтому по ключу можно за доли секунды получить все атрибуты файла.

Запрос на получение файлов по 4 атрибутам является одним из самых часто встречающихся запросов. При поиске DICOM файлов параметрами поиска в большинстве случаев служат теги PatientID, StudyInstanceUID, SeriesInstanceUID, SopClassesUID. При использовании SQL нам потребуется 4 операции JOIN для таблицы meta_file_attribute. При использовании схемы Cassandra можно задать параметр, по которому отсеивать файлы. Извлечение будет происходить аналогично, записи, не удовлетворяющие критерию, будут отсеиваться.

В запросе на получение всех файлов, у которых заданный атрибут равен заданному значению, в схеме PostgreSQL будет использован лишь один JOIN для объединения таблиц meta_file и meta_file_attribute. В Cassandra такой запрос будет выполняться аналогично предыдущему случаю.

Рассмотренные запросы являются одними из самых часто используемых при поиске DICOM файлов по метаданным в базе данных. Тем не менее, существует еще множество запросов, которые в данном примере не рассматриваются. Их удельный вес среди выше указанных запросов составляет менее 5%. Некоторые из них в архитектуре NoSQL могут работать значительно медленнее, чем в реляционных базах данных. Тем не менее, одной из ключевых концепций NoSQL систем является избыточность данных. В большинстве случаев можно, не убирая существующие данные, добавить новые записи таким образом, чтобы различные запросы проходили за минимальное время. Очевидно, что для этого требуется дополнительное пространство на накопителях.

Следует отдельно отметить, что в NoSQL СУБД запросы на обновление данных выполняются дольше, чем в реляционной СУБД. Это обуславливается необходимостью синхронизации всех распределенных хранилищ с обновленными данными. В примере с DICOM файлами очевидно, что атрибуты файла статичны и изменению в базе данных в дальнейшем подлежать не будут.

Выводы

NoSQL системы управления базами данных являются мощным инструментом для разработки больших распределенных систем. При правильном проектировании архитектуры системы и наличии достаточных аппаратных ресурсов, с использованием NoSQL можно добиться увеличения скорости работы с базой данных в десятки раз.

В то же время нереляционные СУБД подходят не для всех типов задач. В проблемах, при решении которых часто приходится обновлять данные в таблицах, а также со сложной логической структурой данных, проектирование правильной NoSQL архитектуры может занять значительное время, а в некоторых случаях вообще оказаться нецелесообразным.

Благодаря специфике хранения данных, NoSQL СУБД в отдельных видах запросов может показывать значительную производительность. Разработчики NoSQL систем рекомендуют при разработке приложений большинство промежуточных данных, которые представляют собой сгруппированные по каким-либо критериям исходные данные, но еще не полностью сформированные результаты, хранить в базе данных, тем самым обеспечивая повышение быстродействия в схожих запросах.

Также при рассмотрении разработок с использованием NoSQL СУБД необходимо учитывать простоту горизонтального масштабирования, и в тоже время ее прозрачность. В каждый момент времени легко разобраться, что происходит в системе и какие узлы участвуют в формировании ответов на запросы.

На сегодняшний день в системах PACS NoSQL еще не применяется. Это обуславливается относительной новизной технологии, а также тем, что на данном этапе развития NoSQL хранилища не всегда могут гарантировать сохранность всех данных, что может быть критично в медицинских системах.

Новые NoSQL проекты и решения предлагаются крупными компаниями, которые используют такие разработки, в первую очередь, в своих целях. Поэтому данные технологии будут активно развиваться с ростом числа пользователей Интернет. На сегодняшний день это перспективное направление исследований, так как оно может помочь многим разработчикам мира отойти от логичной и последовательной реляционной концепции и прийти к избыточности данных, нарушению целостности, но, вместе с тем, к значительному приросту производительности и скорости.

Литература

1. Clunie, David A.; DICOM Structured Reporting; PixelMed Publishing, Bangor, PA, 2001.
2. Revet, Bas; DICOM Cook Book for Implementations in Modalities; 1997; Philips Medical Systems.
3. International Telecommunication Union [Electronic resource] / Интернет-ресурс. - Режим доступа : FTP: <ftp://medical.nema.org/medical/dicom/2009/>
4. International Telecommunication Union [Electronic resource] / Интернет-ресурс. - Режим доступа : www/URL: <http://wiki.apache.org/cassandra/> - Cassandra Wiki
5. International Telecommunication Union [Electronic resource] / Интернет-ресурс. - Режим доступа : www/URL: http://wiki.postgresql.org/wiki/Main_Page - PostgreSQL Wiki
6. International Telecommunication Union [Electronic resource] / Интернет-ресурс. - Режим доступа : www/URL: <http://jdevnotes.blogspot.com/2010/12/apache-cassandra.html> - Java Dev Notes
7. International Telecommunication Union [Electronic resource] / Интернет-ресурс. - Режим доступа : www/URL: <http://wiki.apache.org/cassandra/CassandraCli> - CassandraCLI - Cassandra Wiki.

УДК 004.272

**АНАЛИЗ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ ТРЕХМЕРНОГО МОДЕЛИРОВАНИЕ
НА ПРИМЕРЕ СОЗДАНИЯ УЧЕБНОЙ МОДЕЛИ ШАХТЫ****Кулибаба С.В., Аноприенко А.Я.**

Донецкий национальный технический университет

Кафедра компьютерной инженерии

E-mail: serzh@mail.ua

Аннотация

Кулибаба С.В., Аноприенко А.Я. Анализ современных технологий трехмерного моделирование на примере создания учебной модели шахты. В данном докладе рассматриваются проблемы и способы построение для учебных целей 3D модели шахтной выработки и моделирование возникновения пожара в ней, с элементами интерактивного воздействия на происходящие события. Рассматриваются существующие программные продукты, и сравнение их функциональности.

Произведена предварительная разработка программы-модели шахты.

Анализируется возможность усовершенствования и увеличения функциональности программы.

Общая постановка проблемы

В современном мире существует множество процессов, которые невозможно или очень сложно описать по той или иной причине на бумаге. Для этого используются компьютерные модели, которые в последнее время занимают все большее место как на производстве, так и в учебных целях.

3D графика – это создание модели используя компьютерные программы. Создается с чертежей, подробных описаний или любой другой точной информацией. В специальных программах модель можно рассмотреть с разных ракурсов, поставить в любые условия окружения и пространства. 3d моделирование и 3d визуализация может быть любой сложности.

Можно создать простую трехмерную модель, с низкой детализацией и упрощенной формы. Или же это может быть более сложная модель, в которой присутствует проработка самых мелких деталей, фактуры, использованы профессиональные приемы (тени, отражения, преломление света и так далее). Конечно, это всерьез влияет на стоимость готовой трехмерной модели, однако позволяет расширить применение трехмерной модели.

Задача системы обучения техники безопасности в горных выработках является актуальной. Раньше она решалась посредством создания макетов выработки и шахты в целом, что является очень трудоемкой и дорогостоящей работой. В связи с повсеместным распространением компьютеров и наличием мощного программного обеспечения для создания 3D-моделей, принято решение разработать систему для наглядного представления студентам горной выработки и связанными с ней объектов. Данная модель должна выполнять как экскурсионные задание (показывать месторасположение основных объектов горной выработки), так и динамические процессы в шахте (например пожар), на которые пользователь может соответственно реагировать.

Выбор среды для разработки 3D модели

Для нахождения оптимальной для заданных целей программы, было изучено множество аналогичных программных продуктов, начиная от простых и бесплатных, заканчивая профессиональными пакетами, стоимостью более 1000\$.

Таблица 1 – Сравнение программ-сред моделирования

Название	Лицензия	Преимущества	Недостатки
Autodesk 3ds max	Платная	Самая распространенная и мощная профессиональная программа. Множество готовых текстур и моделей	Очень ресурсоемкая. Платная.
Autodesk Maya	Платная	Одна из самых распространенных и удобных систем моделирования	Платная
Google SketchUP	Бесплатная	Самое большое количество готовых моделей. Бесплатная.	Небольшой функционал
Blender 3D	Бесплатная	Бесплатная. Активно развивающаяся программа, имеющая множество уникальных функций.	Сложный интерфейс управления и моделирования.

Предварительное моделирование системы

Исходя из того, что разрабатываемая система не является коммерческой, то использование платных программных продуктов не допустимо. Вся работа выполнена с использованием Google SketchUP и Blender, посредством импорта/экспорта моделей между системами моделирования. При этом Google SketchUP используется, как огромная библиотека моделей и для создания новых объектов, а Blender – для создания динамических сцен, с возможностью пользователя влиять на происходящие процессы.

Схема модели выбрана на основании реальной учебной системы тушения очага пожара. Предварительный план модели моделирования приведен на рис 1. Основная задача системы заключается не просто в построении компьютерной системы штрека шахты, а создание интерактивной обучающей системы, включающей в себя текстовое и/или звуковое сопровождение происходящих событий, управление камерой, средствами тушения пожара, и др.

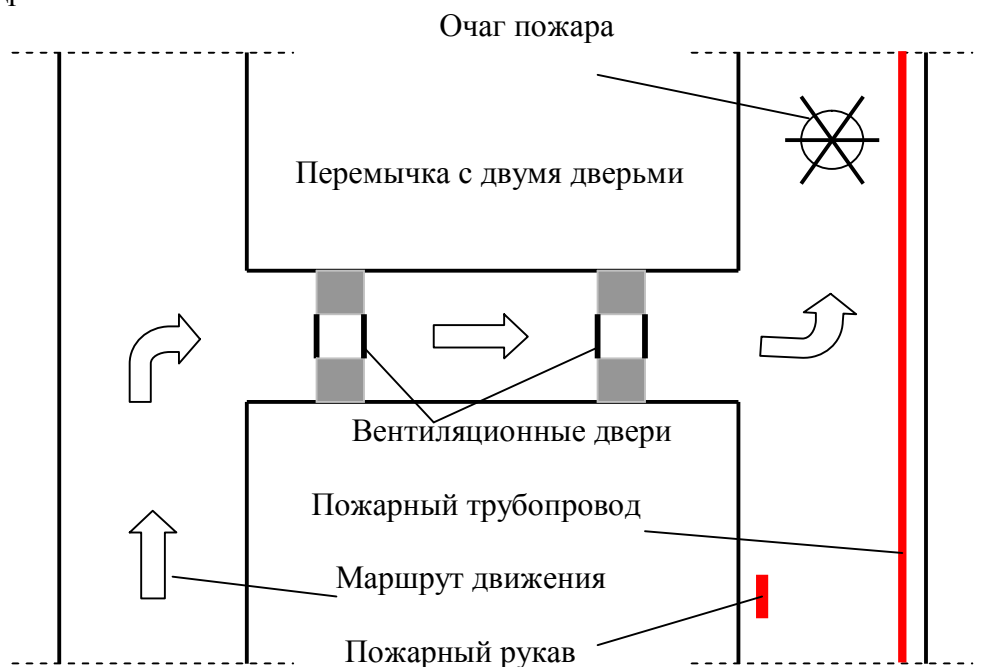


Рисунок 1- Схема модели и маршрута движения модели

В предварительной модели, выполненной средствами ПО Blender 3D присутствует полная модель представленной схемы. Интерактивность заключается в управлении камерой (зрителем) с помощью клавиатуры и мыши, при этом, для открытия дверей, находящихся в переходе между забоями используется клавиша «Ввод». Все размеры модели и текстуры выполнены на основании фотоснимков шахтных выработок, что позволяет получить более точную картину происходящих процессов, а следовательно, приблизиться к главному требованию – реалистичности.

Данная обучающая система находится только на начальном уровне проектирования, так как не учитывает физических параметров, не достаточно реалистична, не реализовано выполнение заданных событий, таких, как:

- Модель очага пожара.
- Видеоинструкции действий при возникновении пожара в шахте.
- Возможность воздействия на очаг возгорания, и в зависимости от действий – изменение силы пламени, перенос его на соседние объекты.
- Виртуальный диалог с диспетчером.
- Визуализация работы пожарной бригады.

В связи со сложностью разработки 3D моделей с помощью программных средств Blender 3D, проведено исследование на возможность импорта объектов из других, более удобных и функциональных систем, таких, как Autodesk 3ds max, Autodesk 3ds Maya и Google SketchUP. Обмен между этими средами моделирования выполняется посредством конвертирования в формат COLLADA, разработанный специально для обмена между 3D приложениями. COLLADA использует открытый стандарт XML для обмена форматами, которые в противном случае были бы несовместимы.

Выводы

Так как реализация 3D модели является очень сложным и трудоемким процессом, то было определено, что вся реализация поставленной задачи должна быть выполнена в существующей среде моделирования. Произведено сравнение программных продуктов создания объемных объектов, и выбраны некоторые из них, которые отвечали поставленным требованиям, таких как: доступность, простота использования, наличие готовых моделей, функциональность.

Проведена оценка сложности задачи, и предварительная реализация 3D модели шахтной выработки в момент возникновения пожара в одной из ее частей. Реализовано интерактивный клип, с возможностью дополнения его новыми событиями, которые необходимо выполнить по условию.

Литература

1. Autodesk - 3D Design [Electronic resource] / Интернет-ресурс.
Режим доступа : www/ URL: <http://www.usa.autodesk.com>. - Загл. с экрана.
2. BlenderУкраина [Electronic resource] / Интернет-ресурс.
Режим доступа : www/ URL: <http://blender.org.ua> - Загл. с экрана.
3. Google SketckUP [Electronic resource] / Интернет-ресурс.
Режим доступа : www/ URL: <http://sketchup.google.com> - Загл. с экрана.
4. Matt Lombard SolidWorks 2010 Bible [Текст] / Wiley, 2010. – С. 10-15.

УДК 004.896

СИНТЕЗ ПРОЕКТОВ ЛОКАЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ С ИСПОЛЬЗОВАНИЕМ ЭКСПЕРТНЫХ МЕТОДИК

Крылов К.С., Григорьев А.В.

Донецкий национальный технический университет
кафедра прикладной математики и информатики
E-mail: krilov.c@gmail.com grigorie@pmi.dgtu.donetsk.ua

Аннотация

Крылов К.С., Григорьев А.В. Синтез проектов локальных вычислительных сетей с использованием экспертных методик. Рассмотрены проблемы использования экспертных систем для синтеза локальных вычислительных систем. Определены проблемы взаимодействия инженера по знаниям и эксперта предметной области. Предложены способы решения существующих проблем наполнения и работы с базами знаний.

Общая постановка проблемы

В настоящее время локальные вычислительные сети (ЛВС) нашли широкое применение во всех сферах жизни человека. Объединение рабочих мест с помощью средств передачи данных позволяют значительно повысить эффективность работы, а относительная простота эксплуатации и гибкость настройки, значительно повлияла на их распространение.

Проектирование ЛВС для небольшого офиса не представляет трудностей, но проектирование ЛВС под большое количество рабочих мест является сложной инженерной задачей. При проектировании необходимо учитывать ряд взаимосвязанных характеристик. Необходимо обеспечить стабильную работу большого количества оборудования, зачастую разных марок[1].

Практически единственным подходом к решению задачи проектирования ЛВС является опыт инженеров. Так как ЛВС являются весьма сложными системами, это часто приводит к принятию неоптимальных решений при их проектировании.

Единственный способ избежать указанных трудностей - использовать средства автоматизированного проектирования. Для автоматизации проектирования, кажется оптимальным использование экспертных систем на основе искусственного интеллекта. Они могут частично заменить специалиста-эксперта при выполнении проектировочных работ, и значительно сократить время разработки проекта.

Исследования в данной области

Существует большое количество готовых «пустых» экспертных систем. Но они более ориентированы на диагностический режим работы, и для начала практического использования, их необходимо наполнить «знаниями». Записи ведутся с использованием языков позволяющих записать на языке предикатов факты и правила логического вывода[2]. Но из-за сложности программирования, сам эксперт предметной области, не сможет наполнить базу «знаниями».

Данную проблему возможно решить при использовании специальной среды SprutExPro (рис 1). Среда позволяет непрограммирующему пользователю создавать прикладные программные комплексы, без привлечения профессиональных программистов. В системе используется метод экспертного программирования. Для записи знаний применяется простейший язык, основанный на правилах, записываемых с помощью ограниченного естественного языка[3].

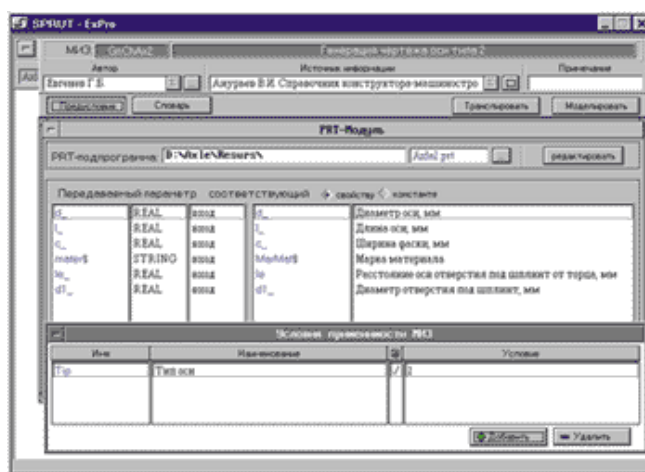


Рисунок 1 – Среда SprutExPro

Другие решения представлены визуальными редакторами, в которых все действия выполняются вручную инженером-проектировщиком (разметка плана, подбор оборудования и т.д.).

Из готовых решений наиболее соответствует предоставленным требованиям и позволяющий автоматизировать процесс проектирования ЛВС онлайн конфигуратор NetWizard. В нем реализовано диалоговое общение с пользователем. Имеется база данных с оборудованием. При работе происходит генерация проекта (рис.2) под конкретные запросы пользователя[4].

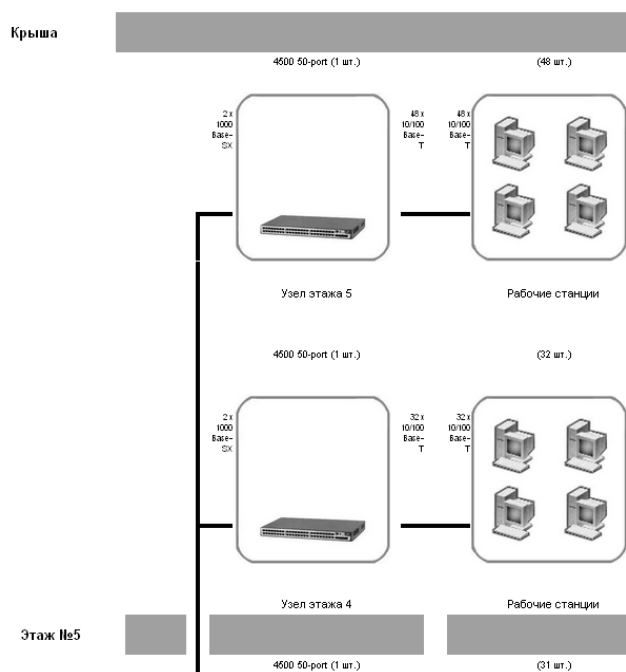


Рисунок 2 – Образец генерации ЛВС

Главным недостатком является закрытость базы оборудования, и проблематичность работы при отсутствии подключения к сети Internet.

Проблемы

Для создания базы, кроме эксперта предметной области, необходим инженер по знаниям. Он выступает посредником между экспертом по знаниям и базой знаний (рис 3).

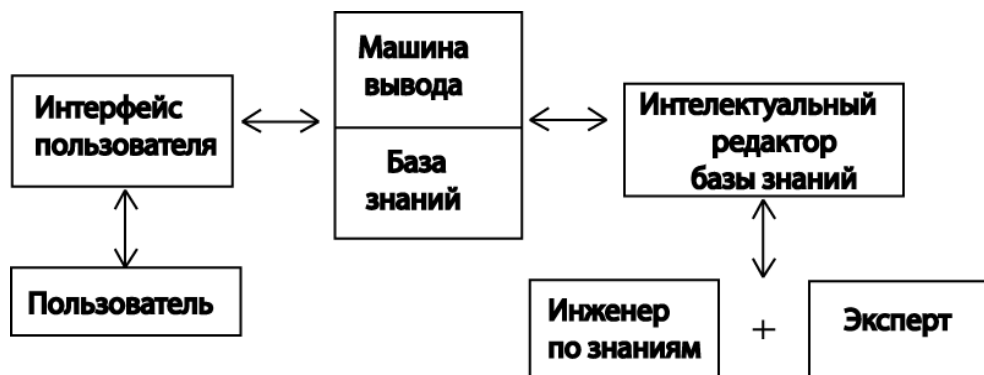


Рисунок 3 – Структура экспертной системы

Возможны три режима взаимодействия инженера по знаниям с экспертом: протокольный анализ, интервью и игровая имитация профессиональной деятельности. Протокольный анализ заключается в фиксации (например, путем записи на магнитную ленту "мыслей вслух" эксперта во время решения проблемы и в последующем анализе полученной информации)[5].

В режиме интервью инженер по знаниям ведет с экспертом активный диалог, направляя его в нужную сторону. Но известно, что потери информации при разговорном общении велики (рис. 4).

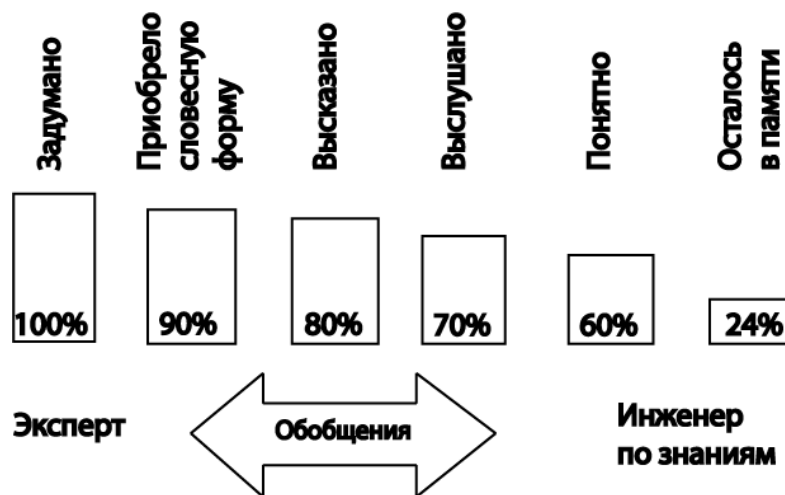


Рисунок 4 – Потери информации при разговорном общении

При игровой имитации эксперт помещается в ситуации, похожие на те в которых протекает его профессиональная деятельность. Наблюдая за его действиями в различных ситуациях, инженер по знаниям, формирует свои соображения об экспертных знаниях, которые впоследствии могут быть уточнены с экспертом в режиме интервью[6].

Каждый из упомянутых способов извлечения знаний имеет свои преимущества и недостатки. Так, при анализе протоколов инженеру по знаниям нелегко отделить понятия, важные для включения в словарь предметной области, от тех, которые при "мыслях вслух" появляются случайно. Кроме того, в протоколах обнаруживаются пробелы, когда рассуждение эксперта как бы прерывается и продолжается уже на основе пропущенных шагов вывода. Заполнение подобных пробелов возможно лишь в режиме интервью. Но и при использовании наиболее оптимальных механизмов извлечения знаний, наполнение базы

может занять значительное время.

Цели исследования

Основной целью является упрощение и ускорение разработки проекта ЛВС, а также облегчение работы инженера. Главная идея заключается в совмещении инженера и эксперта по знаниям. Это уменьшит временные потери при наполнении базы знаний, и позволит более гибко и эффективно использовать опыт и знания эксперта. Для этого необходимо разработать редактора баз знаний с интуитивно понятным интерфейсом. Данный редактор должен позволять наполнять и редактировать базу знаний специалистам не обладающих специфическими знаниями. Обучение работе с редактором должно составлять минимальное время.

Результаты исследований

Экспертная система ориентирована в первую очередь на специалистов предметной области. Быстрое развитие отрасли и появление новых моделей оборудования ставят требованием постоянное редактирование базы «знаний» для получения наиболее оптимальных результатов. При наличии редактора, не требующего специфичный навыков работы с ним, каждый эксперт работающий с данной системой может вносить либо изменять «знания» в общей базе.

При работе системы ставится цель построить оптимальную модель локальной вычислительной сети. В ней должны присутствовать определенные компоненты сети, соединенные связями по заданным правилам. Определение полного перечня конкретного оборудования, необходимого для построения сети, делается в два этапа (две фазы).

На первом этапе с помощью диалога составляется набор правил и требований. На основе их строится модель сети, в которой определяются число узловых элементов модели и параметры связей, объединяющих эти узловые элементы. Узловые элементы на первом этапе представляют собой пустые значения. На втором этапе построения модели происходит "наполнение" узловых элементов реальным оборудованием на основе параметров, определенных в первой фазе с использованием экспертных методик.

Такое двухэтапное построение модели локальной вычислительной сети позволяет на первом этапе абстрагироваться от выбора реального оборудования. Для оптимизации работы системы, необходимо использовать модульную базу знаний (рис. 5).

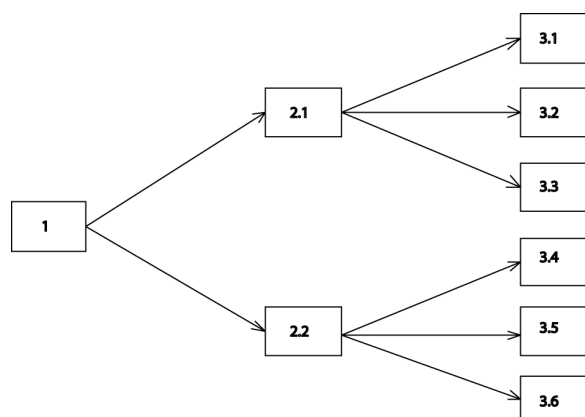


Рисунок 5 – Структура модулей.

Преимущества модульного принципа построения базы знаний состоят в следующем:

- модули обеспечивают средства и методы для сохранения базы знаний в отдельно хранимых файлах;
- разбиение базы знаний на модули учитывается механизмом вывода в процессе работы;

- модули могут активироваться и деактивироваться, что позволяют исключить из рассмотрения правила неактивных модулей.

Модули базы знаний могут быть организованы в иерархию (модуль может быть модулем верхнего уровня или подчиненным модулем). Таким образом, кроме удобства построения и восприятия, модульная база знаний удобна и в процессе использования. При работе системы значительно повышается производительность механизма логического вывода[7].

При работе экспертной системы формируется структурная схема локальной вычислительной сети, в состав которой входят пассивное и активное сетевое оборудование. Также в табличном виде предоставляется конфигурация серверов и рабочих станций с необходимым программным обеспечением.

Если инженера не удовлетворяет полученный результат, он имеет возможность изменить базу знаний с помощью редактора знаний, без привлечения инженера по знаниям. Редактор должен обеспечивать интерпретацию знаний специалиста.

Выводы

В данной работе рассмотрены проблемы использования экспертных методик для проектирования локальных вычислительных систем.

При использовании готовых «пустых» экспертных систем, главной проблемой является наполнение их знаниями. Для этого приходится привлекать инженеров по знаниям, так как эксперты предметной области могут не обладать требуемыми познаниями для заполнения базы знаний.

Предложенные решения позволяют использовать экспертные системы без специфичных навыков построения баз знаний. Подход, основанный на модульном принципе построения базы знаний экспертной системы, может быть использован в аналогичных задачах проектирования, где необходим синтез схем.

Использование удобного редактора позволит создавать и редактировать базы знаний экспертам различных предметных областей без привлечения инженеров по знаниям.

Литература

1. Олифер, Н. А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 3-е издание / В. Г. Олифер // СПб.: Питер, 2006 –С. 45-79.
2. Джарратано Д. С. Экспертные системы: принципы разработки и программирования. 4-е издание/ Д. С. Джарратано, Г. Д. Райли // Пер. с англ. – М.: «Вильямс», 2003 –С. 551-552.
3. Sprut ExPro – СПРУТ-Технология / Интернет-ресурс. – Режим доступа: [www/URL: http://www.sprut.ru/productsandservices/tools/sprutepro](http://www.sprut.ru/productsandservices/tools/sprutepro). - Загл. с экрана.
4. NetWizard / Интернет-ресурс. – Режим доступа: [www/URL: http://netwizard.ru/](http://netwizard.ru/). – Загл. с экрана.
5. Люгер, Д. Ф. Искусственный интеллект: стратегии и методы решения сложных проблем. 4-е издание / Д. Ф. Люгер // Пер. с англ. – М.: «Вильямс», 2003 -С. 219-262.
6. Гаврилова, Т. А. Базы знаний интеллектуальных систем / Т. А. Гаврилова, В. Ф. Хорошевский // СПб.: Питер, 2000 –С. 71-73.
7. Ильянсов Б. Г. Модульная структура базы знаний экспертной системы «Выбор специальности при поступлении в вуз»/ Б. Г. Ильясов, Е. Б. Старцева, Н. Р. Янгуразова// Электронный журнал Нефтегазовое дело, 2006 –С. 4-5.

УДК 004.75

**ОБ ИССЛЕДОВАНИИ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ПОСТРОЕНИЯ
МАРКОВСКИХ МОДЕЛЕЙ ВЫЧИСЛИТЕЛЬНЫХ КЛАСТЕРОВ****Михайлова Т.В., Абдулина О.Р.**

Донецкий национальный технический университет

Кафедра прикладной математики и информатики

E-mail: oksufka@gmail.com**Аннотация**

Михайлова Т.В., Абдулина О.Р. Об исследовании параллельных алгоритмов построения марковских моделей вычислительных кластеров. Рассмотрены параллельные алгоритмы построения дискретной модели Маркова однородного и неоднородного кластеров.

Введение

Исследование современных сложных моделей требует использования мощных вычислительных систем, в частности, кластеров (групп компьютеров, соединенных скоростными каналами связи). Для исследования эффективности использования кластеров в настоящее время разрабатываются параллельные алгоритмы построения марковских моделей кластеров. Ссылки на основные работы в этой области приводятся в статьях [1-6].

Для исследования эффективности функционирования кластеров и выработки рекомендаций рационального использования ресурсов могут быть применены непрерывные или дискретные аналитические модели. Дискретная модель по сравнению с непрерывной – более точная. Однако анализ кластерных систем с помощью дискретных моделей при большом количестве решаемых задач на ЭВМ сопряжен с большими временными затратами, так как количество состояний дискретной марковской модели комбинаторно возрастает при увеличении количества задач. Для уменьшения временных затрат в работе [1] предлагается распараллелить алгоритм построения дискретной модели кластера и оценить трудоемкость этого алгоритма и характеристики результатов распараллеливания.

Дискретная модель кластера с совместным разделением дискового пространства при использовании методов построения дискретных Марковских моделей описана в [4]. Анализ кластерных систем с помощью этих моделей при большом количестве решаемых задач на ЭВМ требует больших временных затрат, так как количество состояний дискретной Марковской модели комбинаторно возрастает при увеличении количества задач. Для решения этой проблемы предложен параллельный алгоритм построения дискретной модели [4].

Эффективность применения кластерных систем в настоящее время наталкивается на целый ряд нерешенных проблем, касающихся как их программного обеспечения, так и, собственно, организации параллельных вычислений и эффективности их выполнения. Объединение большого количества компьютеров в единый вычислительный комплекс породило такие задачи, как сбалансированность нагрузки, выявление узких мест в вычислительной системе, обеспечение заданного времени ответа при работе большого количества пользователей, выбор оборудования при максимальной производительности при ограниченной заданной стоимости и т.д.

В связи с большим количеством создаваемых параллельных вычислительных систем и, во многих случаях, неэффективным их использованием, работы в этом направлении требуют дальнейшего развития и не утрачивают своей актуальности.

Дискретная модель Маркова однородного кластера

Однородные кластеры с совместным использованием дискового пространства используют СУБД Oracle Parallel Server и Informix.

Структурная схема марковской модели кластера с совместным использованием дискового пространства приведена на рис. 1 [1]. Количество серверов-приложений – $N1$. Каждый из них может обратиться к данным, расположенным на дисках, количество которых $N2$. Ввиду ограниченных вычислительных возможностей будем считать, что количество задач, обрабатываемое такой вычислительной системой, не более M .

Допустим, задачи, обрабатываемые на таком кластере, однородные и имеют следующие характеристики: $p12$ – вероятность запроса к одному из $N1$ серверов, $p23$ – вероятность запроса к одному из $N2$ дисков, $p21$ – вероятность завершения обслуживания одним из $N1$ серверов, $p10$ – вероятность завершения обслуживания задачи, $q0$ – вероятность появления новой задачи.

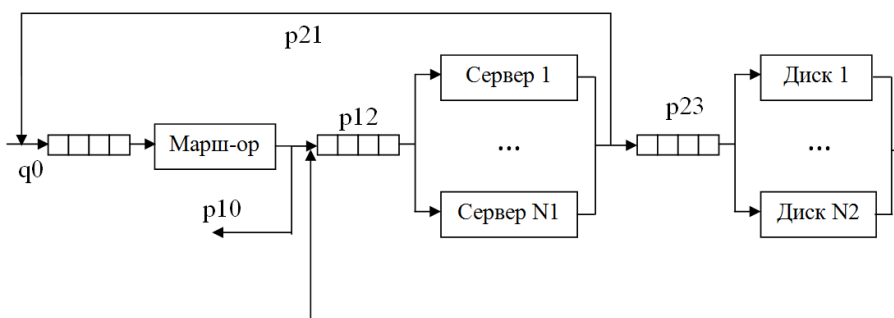


Рис. 1 – Структурная схема Марковской модели кластера с совместным использованием дискового пространства

Дискретная марковская модель неоднородного кластера.

Упрощенная модель кластера с совместным использованием дискового пространства представлена на рис. 2 [6].

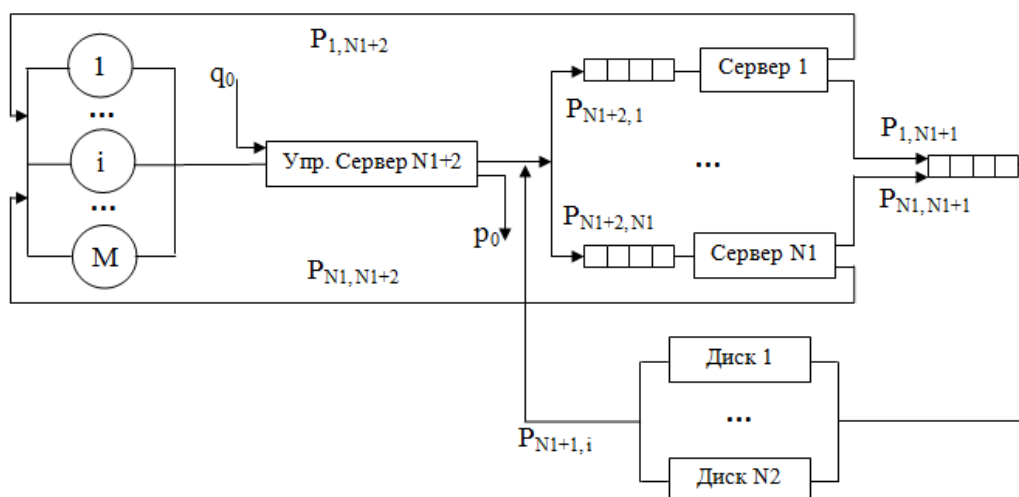


Рис. 2. Структурная схема марковской модели неоднородного кластера с совместным использованием дискового пространства

Управляющий сервер распределяет между, выполняющими неодинаковые приложения, серверами приложений задачи. При этом, количество серверов приложений было принято за $N1$, а количество дисков, содержащих данные, к которым обращаются

сервера-приложения, N_2 . Такая вычислительная система обрабатывает не более M задач, если принять во внимание то, что вычислительные возможности ограничены.

Требования, поступающие на обслуживание на каждый из серверов, поступают в соответствующую ограниченную очередь (не более M), из которой на обслуживание выбираются по правилу «первый пришел – первый обслужился».

Характеристики неоднородных задач, обрабатываемых на таком кластере:

$P_{N_1+2,i}$ – вероятность запроса к i -му серверу, $i=1..N_1$, P_{i,N_1+1} – вероятность запроса i -го сервера к одному из N_2 дисков, $i=1..N_1$, P_{i,N_1+2} – вероятность завершения обслуживания задачи i -м сервером, $i=1..N_1$, $P_{N_1+1,i}$ – вероятность завершения обслуживания задачи одним из N_2 дисков и поступление на i -й сервер, $i=1..N_1$, $P_{N_1+2,0}$ – вероятность завершения обслуживания задачи, P_{0,N_1+2} – вероятность появления новой задачи.

Однако с увеличением количества состояний эта и подобные модели не считаются на традиционных ЭВМ. Дискретная марковская модель кластера легко распараллеливается на различные параллельные структуры.

Стационарные вероятности дают возможность определить средние значения временных характеристик обслуживания и занятости устройств вычислительной системы, а именно: среднее число занятых устройств в s -м узле, загрузка устройств, среднее число задач в s -м узле, среднее число задач, находящихся в очереди к s -му узлу, среднее время пребывания в s -м узле, среднее время ожидания в s -м узле, среднее время пребывания в вычислительной системе, среднее время ожидания в вычислительной системе.

Параллельные алгоритмы разбиваются на две составляющие. Первая представляет собой параллельный алгоритм построения матрицы переходных вероятностей, каждый элемент которой вычисляется независимо от остальных. Во второй части вычисляется вектор стационарных вероятностей.

Эффективность параллельного алгоритма вычисления матрицы переходных вероятностей с увеличением количества решаемых задач M приближается к единице (рис. 3), а с увеличением количества процессоров незначительно уменьшается (это объясняется наличием простаивающих процессоров).

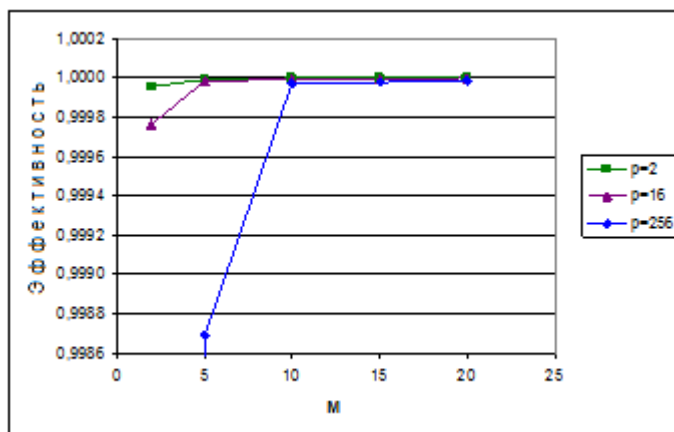


Рис. 3. Эффективность параллельного алгоритма вычисления матрицы переходных вероятностей в зависимости от количества решаемых задач M на разном количестве процессоров

Зависимость ускорения от количества обрабатываемых на кластере задач, приближается к количеству процессоров, и графически представлена на рис. 4 а) для количества процессоров $p=2$, на рис. 4 б) для $p=16$, на рис.4 в) для $p=256$.

Однако полученные матрицы переходных вероятностей сильно разрежены и требуют применения специальных алгоритмов обработки, что необходимо учесть при реализации этих моделей.

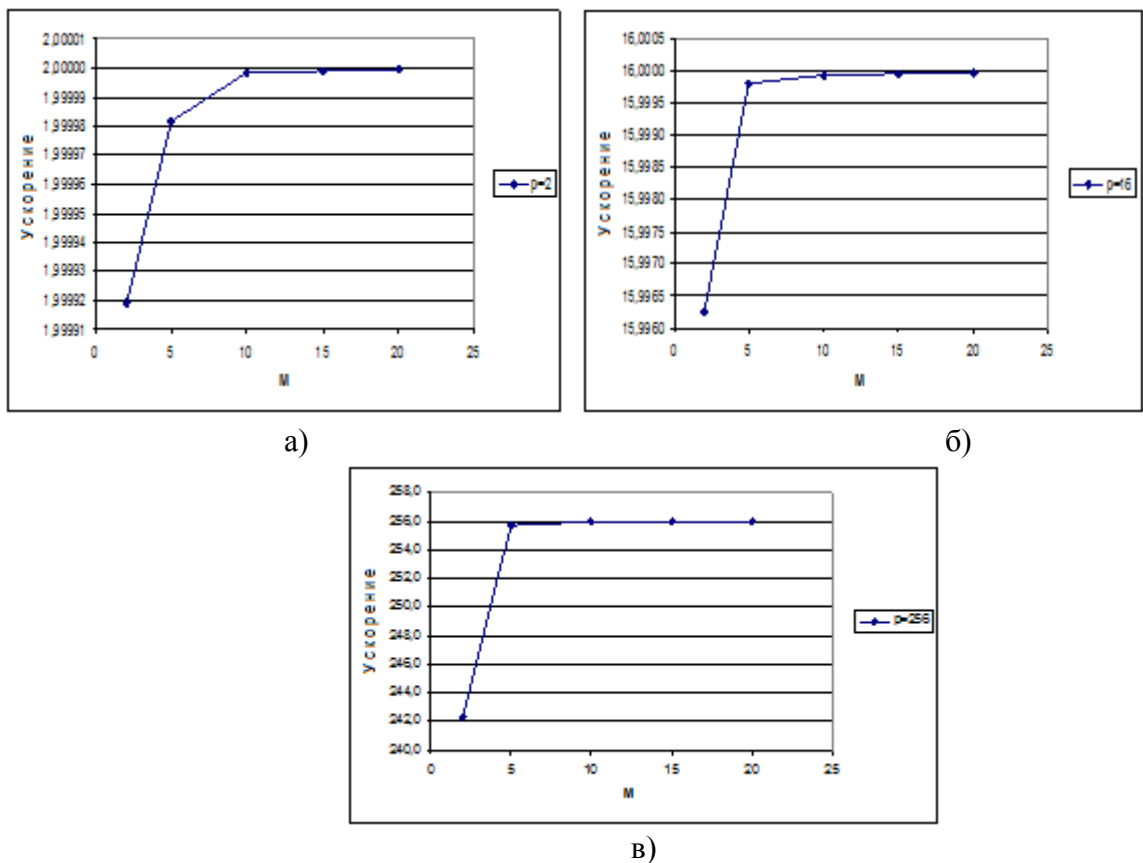


Рис. 4. Ускорение параллельного алгоритма вычисления матрицы переходных вероятностей в зависимости от количества решаемых задач M. Количество процессоров а) $p=2$, б) $p=16$, в) $p=256$.

Для того, чтобы сбалансировать загрузку вычислительных модулей кластера, в качестве базовой подзадачи берется расчет строки, столбца или блока матрицы переходных вероятностей. Каждый процесс определяет, какие строки (столбцы, блоки) матрицы переходных вероятностей он должен вычислить и отправить управляющему процессу. Схема параллельного алгоритма вычисления элементов матрицы переходных вероятностей приведена на рис. 5.



Рис. 5. Параллельный алгоритм построения матрицы переходных вероятностей

Расчет стационарных вероятностей можно реализовать с использованием итерационного алгоритма, в котором в качестве базового используется алгоритм умножения матрицы на вектор. Схема алгоритма расчета вектора стационарных вероятностей представлена на рис.6. Для распараллеливания задачи умножения матриц можно использовать ленточную (по строкам, по столбцам) или блочную схему разделения данных.

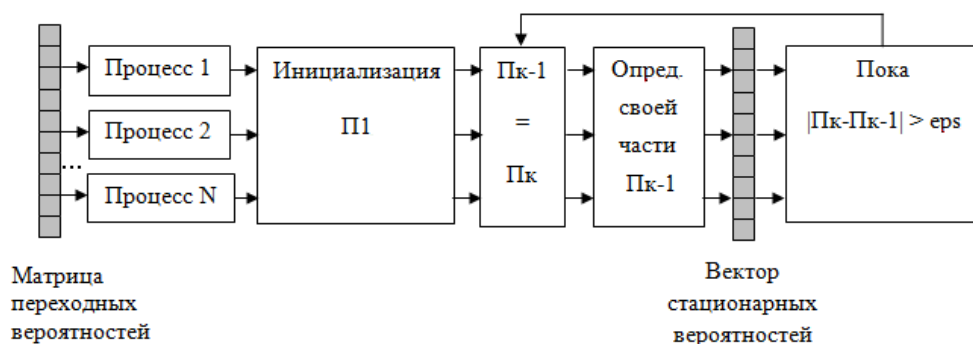


Рис. 6. Параллельный алгоритм расчета вектора стационарных вероятностей

Отметим, что целесообразно определять оптимальное количество ВМ для вычисления параллельного алгоритма, так как оно напрямую зависит от размерности задачи и количества пересылок.

Выводы

Таким образом, предложенные параллельные алгоритмы построения дискретной марковской модели неоднородного кластера, дают возможность оценить эффективность работы однородной и неоднородной параллельных вычислительных систем с совместным использованием дискового пространства любой конфигурации, так как эффективность параллельного алгоритма стремится к единице. Результаты моделирования могут быть использованы в различных направлениях, например для повышения эффективности работы, для анализа и синтеза, реконфигурации кластерных систем подобного типа. В дальнейшем возможно исследование этого алгоритма на параллельных структурах различной топологии.

Литература

1. Фельдман Л.П., Михайлова Т.В. Дискретная модель Маркова однородного кластера / Искусственный интеллект.– Донецк: ИПШ МОН і НАН України “Наука і освіта”, №3, 2006. – С. 79-91.
2. Михайлова Т.В. Оценка точности непрерывной и дискретной моделей Маркова // Научные труды Донецкого государственного технического университета. Серия «Информатика, кибернетика и вычислительная техника» (ИКВТ-2005). – 2005. – Вып. 93. – С. 114-122
3. Фельдман Л.П., Михайлова Т.В., Ролдугин А.В. Реализация параллельного алгоритма построения марковских моделей. // Наукові праці ДонНТУ. Серія: Інформатика, кібернетика та обчислювальна техніка, випуск 10.
4. Михайлова Т.В. Параллельный алгоритм построения дискретной модели Маркова // Искусственный интеллект. Интеллектуальные и многопроцессорные системы. Материалы Международной научно-технической конференции, 25-30 сентября 2006г., Таганрог-Донецк-Минск, 2006.
5. Фельдман Л.П., Михайлова Т.В. Использование аналитических методов для оценки эффективности многопроцессорных вычислительных систем. //Электронное моделирование, Т.29, №2, 2007.- С.17-27
6. Кучереносова О.В. [Параллельный алгоритм построения дискретной марковской модели неоднородного кластера](#) // ДонНТУ. Международная научно-техническая конференция студентов, аспирантов и молодых ученых «Информационные управляющие системы и компьютерный мониторинг – 2010» – Донецк, 2010

УДК 004

ЭФФЕКТИВНОСТЬ И МАСШТАБИРУЕМОСТЬ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ БЛОЧНОГО УМНОЖЕНИЯ ПЛОТНО ЗАПОЛНЕННЫХ МАТРИЦ

Лямина О. В., Назарова И. А.

Донецкий национальный технический университет

Кафедра прикладной математики и информатики

E-mail: liamina_olga@mail.ru

Аннотация

Лямина О.В., Назарова И.А. Эффективность и масштабируемость параллельных алгоритмов блочного умножения плотно заполненных матриц. Рассмотрены параллельные алгоритмы умножения матриц семейства Кэннона. Проведен анализ зависимости времени выполнения и ускорения от числа процессоров и размера матриц. Оценена эффективность алгоритмов.

Постановка задачи

Матричное умножение – одна из основных операций, которая выполняется при решении различных задач: решение системы линейных алгебраических уравнений, дифференциальных уравнений. Умножение матриц является трудоемким с операционной и коммутативной точки зрения. Поэтому эффективность решения этой задачи – важный фактор.

Для эффективного выполнения умножения матриц используются параллельные алгоритмы. Топология решетка и гиперкуб являются наиболее подходящими для реализации таких операций. Для такой топологии вычисление матричных арифметических операций можно свести к выполнению операций с блоками матриц.

Существует несколько таких алгоритмов. В данной работе рассматривается семейство алгоритмов Кэннона, которое основано на блочном разбиении матриц.

В алгоритме Кэннона две исходные матрицы A и B и матрица результат C разделяются на блоки. Семейства Кэннона изменяет отображения блоков двух из трех матриц, которые берут участие в вычислении произведения.

Пусть количество столбцов/строк матрицы n кратно числу узлов решетки p . Количество узлов решетки по вертикали/горизонтали равно q . Если представить матрицы в виде квадратных блоков размером $k = \frac{n}{q}$ элементов, то каждому узлу можно однозначно поставить в соответствие такой блок.

Алгоритм вычисления матричного произведения с сохранением отображения блоков матрицы-результата C .

Алгоритм включает в себя шаги:

- 1) блоки строк матрицы A сдвигаются циклично влево на i узлов по горизонтали, где i - индекс строки матрицы A .
- 2) блоки столбцов матрицы B сдвигаются циклично вверх на j узлов по вертикали, где j – индекс столбца матрицы A .

Алгоритм выполняется за q шагов, где q – размерность вычислительной решетки. Каждый шаг состоит из следующих действий:

- а) на вычислительном узле решетки с индексами (i, j) производится умножение блоков

A_{ij} и B_{ij} .

б) циклическое смещение блоков матрицы A влево на 1 узел по горизонтали решетки.

с) циклическое смещение блоков матрицы B вверх на 1 узел по вертикали решетки.

Результат умножения матриц хранится в матрицы C , блоки которой не подлежат смещению.

Анализ эффективности

Время выполнения п.1) или п.2) согласно [1] можно рассчитать по формуле:

$$T_{alignAB} = (t_s + t_w \cdot k^2)(q+1) \quad (1)$$

где t_s - латентность, t_w - время передачи слова данных.

Время умножения матриц в одном блоке:

$$T_{AB} = (k^2 \cdot (2k-1) + k^2) \cdot \tau \quad (2)$$

Время циклического сдвига для п. с) и б):

$$T_{rollShift} = (t_s + t_w \cdot k^2)(q+1) \quad (3)$$

Суммарное время выполнения алгоритма:

$$T_{CannonC} = 2qT_{alignAB} + 2T_{rollShift} + pT_{AB} \quad (4)$$

$$T_{CannonC} = (2q+2)(t_s + t_w \cdot k^2)(q+1) + (k^2 \cdot (2k-1) + k^2) \cdot p \cdot \tau \quad (5)$$

Отсюда получаем ускорение параллельного алгоритма и эффективность использования параллельным алгоритмом процессоров при решении задачи:

$$S_{CannonC} = \frac{(k \cdot q)^2 (2k \cdot q - 1)}{(2q+2)(t_s + t_w \cdot k^2)(q+1) + (k^2 \cdot (2k-1) + k^2) \cdot p \cdot \tau} \quad (6)$$

$$E_{CannonC} = \frac{S_{CannonC}}{p} \quad (7)$$

Алгоритм вычисления матричного произведения с сохранением отображения блоков матрицы A .

Алгоритм включает в себя шаги:

1) блоки строк матрицы B сдвигаются циклично вправо на i узлов по горизонтали, где i - индекс строки матрицы B .

2) блоки столбцов матрицы B сдвигаются циклично вверх на j узлов по вертикали, где j - индекс столбца матрицы A .

3) блоки строк матрицы C сдвигаются циклично вправо на i узлов по горизонтали, где i - индекс строки матрицы C .

Алгоритм выполняется за q шагов, где q - размерность вычислительной решетки.

Каждый шаг состоит из следующих действий:

а) на вычислительном узле решетки с индексами (i, j) производится умножение блоков A_{ij} и B_{ij} .

б) Циклическое смещение блоков матрицы C вправо на 1 узел по горизонтали решетки.

с) Циклическое смещение блоков матрицы B вверх на 1 узел по вертикали решетки.

Результат умножения матриц хранится в матрицы C , блоки которой подлежат смещению. Поэтому по завершению нужно выровнять матрицу до выходного отображения блоков.

Анализ эффективности

Время выполнения п.1), п. 2) или п.3) просчитывается по формуле (1). Время умножения матриц в одном блоке – формула (2). Время циклического сдвига для п. с) и б) – (3). После выполнения q шагов матрицы B и C необходимо выровнять до выходного отображения блоков на узле вычислительной решетке. Время выполнения рассчитывается по (1).

Суммарное время выполнения алгоритма:

$$T_{CannonA} = 3qT_{alignAB} + 2T_{rollShift} + pT_{AB}$$

$$T_{CannonA} = (3q + 2)(t_s + t_w \cdot k^2)(q + 1) + (k^2 \cdot (2k - 1) + k^2) \cdot p \cdot \tau \quad (8)$$

Отсюда получаем ускорение параллельного алгоритма и эффективность использования параллельным алгоритмом процессоров при решении задачи:

$$S_{CannonA} = \frac{(k \cdot q)^2 (2k \cdot q - 1)}{(3q + 2)(t_s + t_w \cdot k^2)(q + 1) + (k^2 \cdot (2k - 1) + k^2) \cdot p \cdot \tau} \quad (9)$$

$$E_{CannonA} = \frac{S_{CannonA}}{p} \quad (10)$$

Алгоритм вычисления матричного произведения с сохранением отображения блоков матрицы В.

Алгоритм включает в себя шаги:

- 1) блоки строк матрицы A сдвигаются циклично влево на i узлов по горизонтали, где i - индекс строки матрицы A .
- 2) блоки столбцов матрицы A сдвигаются циклично вниз на j узлов по вертикали, где j – индекс столбца матрицы A .
- 3) блоки столбцов матрицы C сдвигаются циклично вниз на i узлов по горизонтали, где i - индекс столбца матрицы C .

Алгоритм выполняется за q шагов, где q – размерность вычислительной решетки.

Каждый шаг состоит из следующих действий:

- а) на вычислительном узле решетки с индексами (i, j) производится умножение блоков A_{ij} и B_{ij} .
- б) Циклическое смещение блоков матрицы A влево на 1 узел по горизонтали решетки.
- с) Циклическое смещение блоков матрицы C вниз на 1 узел по вертикали решетки.

Результат умножения матриц хранится в матрицы C , блоки которой подлежат смещению. Поэтому по завершению нужно выровнять матрицу до выходного отображения блоков.

Анализ эффективности

Аналогично алгоритму вычисления матричного произведения с сохранением отображения блоков матрицы A рассчитываем суммарное время выполнения алгоритма:

$$T_{CannonB} = 3qT_{alignAB} + 2T_{rollShift} + pT_{AB}$$

$$T_{CannonB} = (3q + 2)(t_s + t_w \cdot k^2)(q + 1) + (k^2 \cdot (2k - 1) + k^2) \cdot p \cdot \tau \quad (11)$$

Получаем ускорение параллельного алгоритма и эффективность использования параллельным алгоритмом процессоров при решении задачи:

$$S_{CannonB} = \frac{(k \cdot q)^2 (2k \cdot q - 1)}{(3q + 2)(t_s + t_w \cdot k^2)(q + 1) + (k^2 \cdot (2k - 1) + k^2) \cdot p \cdot \tau} \quad (12)$$

$$E_{CannonB} = \frac{S_{CannonB}}{p} \quad (13)$$

Сравнительный анализ алгоритмов

Отличия вышерассмотренных алгоритмов состоит в коммуникационных затратах. Рассмотрим графики поведения алгоритмов, на которых черным цветом представлен алгоритм вычисления матричного произведения с сохранением отображения блоков матрицы-результата C , а серым - алгоритмы вычисления матричного произведения с сохранением отображения блоков матрицы A и B .

На Рис.1 отображается поведение функции времени для фиксированных матриц, $n = 10000$, в зависимости от количества узлов решетки.

На Рис.2 показана зависимость функции времени для фиксированного числа узлов решетки, $p = 10000$, в зависимости от размера матриц.

На Рис.3 показано поведение функции ускорения для фиксированных матриц, $n = 10000$, в зависимости от количества узлов решетки.

На Рис.4 показана зависимость функции ускорения фиксированного числа узлов решетки, $p = 10000$, в зависимости от размера матриц.

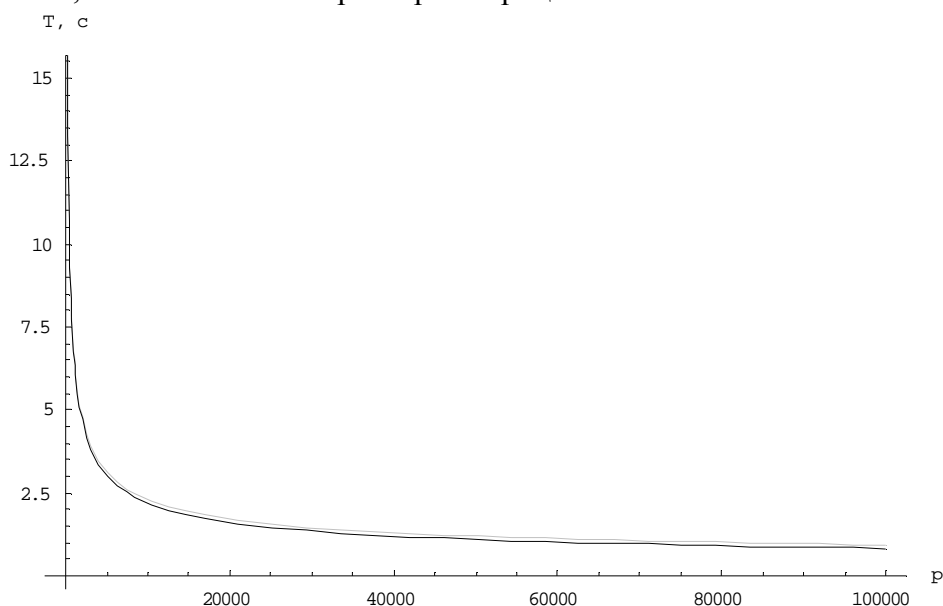


Рис.1 – График зависимости времени выполнения от количества узлов

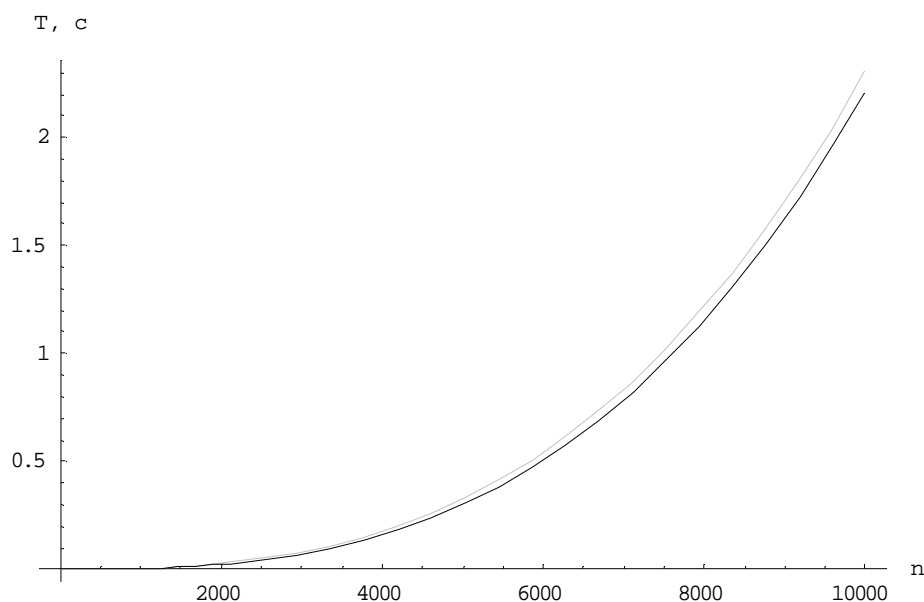


Рис.2 – График зависимости времени выполнения от размера матрицы

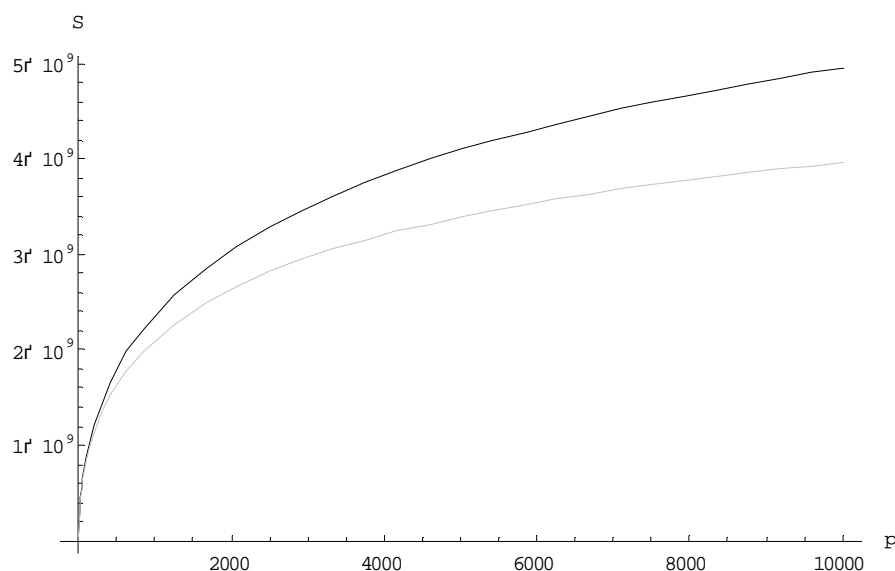


Рис.3 – График зависимости ускорения от количества узлов

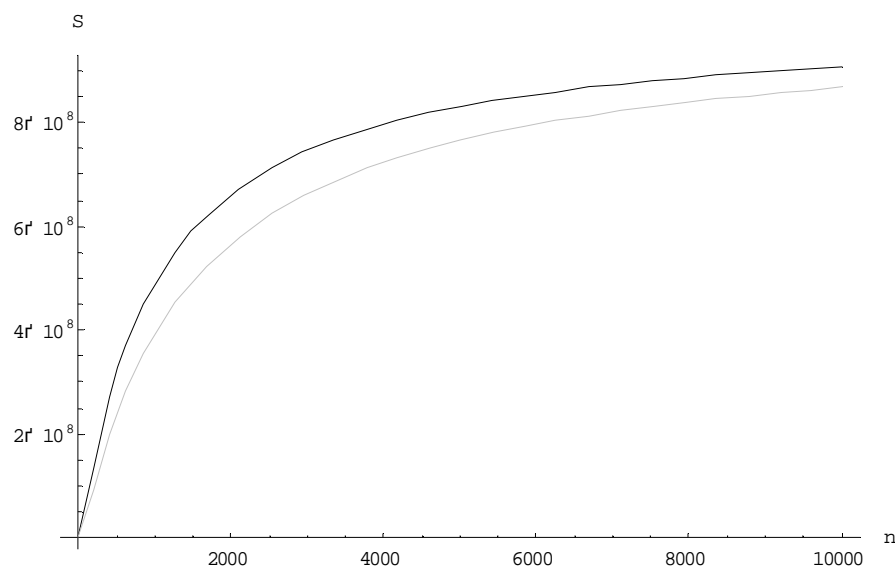


Рис.4 – График зависимости ускорения от размера матрицы

Выводы

В работе были проведены исследования эффективности и масштабируемости алгоритмов семейства Кэннона. Все три алгоритма показывали близкие значения в отдельных ситуациях, однако наиболее эффективным с точки зрения времени выполнения оказался алгоритм вычисления матричного произведения с сохранением отображения блоков матрицы-результата C , что было определено, анализируя графики зависимостей.

Литература

1. Гергель В.П. Теория и практика параллельных вычислений.
2. Интернет-Университет Информационных Технологий - дистанционное образование. <http://www.intuit.ru/>
3. Gupta A., Kumar V. Scalability of parallel algorithm for matrix multiplication // Technical report TR-91-54, Department of CSU of Minneapolis, 2001

УДК 004

**ОРГАНИЗАЦИЯ ПРОЦЕССА ИНДИВИДУАЛИЗИРОВАННОГО ОБУЧЕНИЯ НА
БАЗЕ АДАПТИВНОЙ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ****Смирнов В.А., Фонов А.М.***Донецкий национальный технический университет г. Донецк**Кафедра автоматизированных систем управления*

E-mail: smirnoffvictor@gmail.com

Аннотация

Смирнов В.А., Фонов А.М. Организация процесса индивидуализированного обучения на базе адаптивной системы дистанционного обучения. Адаптивные обучающие системы являются одним из актуальных направлений исследований и разработок в современном электронном обучении. В основе адаптивных механизмов таких систем лежит модель ученика. В статье рассматривается вопрос интеграции моделей знаний ученика из разных курсов.

Общая постановка проблемы. Методика преподавания с использованием технологий дистанционного обучения существенно отличается от традиционных технологий обучения и в основном опирается на самостоятельное изучение курса студентом, причем значительная часть работы преподавателя переводится на ЭВМ. Фактически обучающие системы "приобретают" знания эксперта-преподавателя и "доказывают" их к студенту. Поэтому можно констатировать, что основной особенностью дистанционного образования является предоставление студентам возможности самостоятельно получать необходимые знания, использовать современные информационные технологиями. Возможность индивидуализации обучения является одним из важнейших преимуществ использования информационных технологий в учебном процессе. При существенной разнице в уровне базовой подготовки и индивидуальных способностей студентов одинаковый для всех план учебного процесса, принятый за основу в традиционных системах дистанционного обучения (СДО), является оптимальным в лучшем случае лишь для 15-30% студентов: для одних он слишком напряженный, для других, наоборот, недостаточно интенсивный [6]. В результате не эффективно используются интеллектуальные и материальные ресурсы как индивида, так и общества. Обзор наиболее известных систем широко используемых в современном дистанционном обучении классические подходы, таких как ANGEL, BlackBoard, Desire2Learn, ILIAS, Lotus LearningSpace, Moodle, WebCT, показал, что, как правило, учебный курс, представленный в среде СДО, представляет собой набор статических гипертекстовых документов. Все студенты получают одинаковый материал для изучения без учета их индивидуальных особенностей. Студент не может получить оперативную помощь от учителя или другого студента, как это происходит в обычном учебном классе. Новое исследовательское направление в сфере дистанционного обучения на Web-платформе - это адаптивные и интеллектуальные технологии в обучении. Задача этого направления исследований - включить в дистанционные обучающие системы возможности индивидуализации [1]. С помощью адаптивных и интеллектуальных технологий обучающая система учитывает индивидуальные способности студента, его предыдущие знания, умения. На основе этих данных о студенте процесс обучения проходит для него оптимальным путем. Все это способствует тому, что сегодня существует проблема обеспечения индивидуализированного обучения в дистанционном образовании, которая заключается в разработке методов, технологий и программных средств создания адаптивных систем дистанционного обучения на базе интеллектуальных интернет-технологий [2].

Постановка задач исследования. Целью работы является анализ требований к адаптивным системам непрерывного обучения, а также методам, моделям и технологиям их построения. Это включает анализ существующих методов и технологий интеллектуальных образовательных систем и определение специфических и слабо развитых направлений, которые требуют основательной проработки в контексте непрерывного обучения.

Решение задачи и результаты исследований. Отличие адаптивных гипермедиа-систем от просто гипермедиа-систем заключается в наличии у первых модели пользователя, на основе которой строится адаптация. Чтобы построить и поддерживать актуальную модель пользователя адаптивная система собирает данные из разных источников, включающий неявное наблюдение за действиями пользователя и явное опроса и введение им данных. Модель пользователя и адаптация являются двумя сторонами одного процесса. Количество и качество информации, представленной в модели пользователя, зависит от типа адаптационного эффекта, который стремится предоставить система [3].

Основными характеристиками, которые моделируются и используются адаптивными Web-системами, являются знания, интересы, цели, предпосылки, индивидуальные особенности и контекст работы пользователей. Каждая адаптивная система, как правило, использует некоторое подмножество данного набора.

Модель пользователя зависит от способа моделирования предметной области в системе. Модель знаний пользователя может быть скалярной, представляющий уровень знаний ученика в предметной области единственным значением по некоторой шкале - количественной (например, от 0 до 5) или качественной (например, отлично, хорошо, удовлетворительно, неудовлетворительно). Наиболее распространенной является оверлейная модель знаний ученика. Она представляет знание ученика как подмножество модели предметной области, которая в свою очередь отражает знание предмета на уровне эксперта [5]. Оверлейная модель сохраняет для каждого фрагмента знаний предметной области некоторую оценку знаний пользователя по данному фрагменту. Существует также модель на основе ошибок, которая представляет как корректные знания студента, так и ложные. Моделирование интересов пользователя - новое направление, характерное для Web-систем. Интересы пользователя приобретают большое значение для адаптивных гипермедиа-систем в связи с увеличением объема информации и ростом популярности таких типов информационно-ориентированных систем как энциклопедии, гипертекстовые системы новостей, электронные магазины, музейные гиды и другие системы, в которых доступ к информации мотивируется интересами [7]. Следует отметить, что эти тенденции, а также популяризация конструктивизма в обучении имеют влияние и на образовательные сервисы, и поэтому требуют большего внимания к модели интересов пользователя в образовательном контексте. Характерным для информационно-поисковых и информационно-фильтрующих систем является построение модели интересов на основе ключевых слов. Зато адаптивные гипермедиа системы адаптировали подход на основе семантических понятий к моделированию интересов [8]. В данном случае модель интересов становится очень похожей на оверлейную модель знаний пользователя. Перспективным является синтез подходов на основе ключевых слов с подходом на основе понятий для модели интересов студента. Цели и задачи представляют ближайшую цель работы пользователя в адаптивной системе. Как правило в обучающих системах в этой модели представляется учебная цель. Достижения учебной цели решается автоматическим планированием и определением последовательности учебного курса. Цели ученика могут моделироваться с помощью каталога целей. Этот подход несколько похож на оверлейную модель. Его основой является заранее определенный каталог возможных целей ученика, которые система должна уметь распознавать.

Предпосылки пользователя касающиеся его предыдущего опыта вне ключевой предметной области системы. Сюда следует отнести профессию, должностные обязанности, опыт работы в связанных областях и даже специфический взгляд пользователя на

предметную область [9]. Отметим, что подобные характеристики являются важными в контексте непрерывного обучения, однако до сих пор они использовались в немногих системах.

Модель индивидуальных особенностей подает характеристики, которые представляют индивидуальность пользователя. Сюда относятся психологические и личностные особенности, когнитивные стили и факторы и учебные стили.

Сравнительно новым направлением является моделирование контекста работы пользователя. Сюда следует отнести представление программно-аппаратной платформы пользователя, его географического положения и эмоционального состояния. Модели и механизмы адаптации. Принцип функционирования адаптивной обучающей системы схематически изображены на рис.1.[5]



Рис. 1. Структура адаптивной обучающей системы

Адаптационные возможности системы зависят от применяемых в ней моделей предметной области и модели пользователя. На этапе принятия решений об адаптации избираются конкретные методы адаптации на основе результатов этапа моделирования пользователя с целью улучшить избранные аспекты взаимодействия ученика и системы. Решения об адаптации могут выражаться в таких действиях как: показ справочного окна для помощи пользователю, выполнении задания, реструктурировании гиперпространства, чтобы помочь студенту ориентироваться и передвигаться в нем, предоставление дополнительных

объяснений по некоторому учебном понятию т.д.

Логика адаптационных решений часто подается с помощью набора правил адаптации, которые определяют какой компонент адаптации необходимо выбрать согласно результатам моделирования ученика. В адаптивных гипермедиа-системах эти правила отвечают за адаптивное представление контента и адаптивную навигацию [5]. Для реализации механизмов адаптации используются подходы на основе семантических понятий предметной области и семантической индексации контента, ключевых слов и автоматической индексации контента на базе информационного поиска, а также социальные механизмы, такие как навигация на основе истории и коллективная фильтрация.

Моделирование взаимодействия участников учебного процесса. Модели коллективного взаимодействия в обучении служат для организации сотрудничества через Интернет участников учебного процесса. Сюда следует отнести такие традиционные средства общения как электронная переписка, форумы, блоги, а также wiki-системы.

Кроме общения, коллективное взаимодействие может также выражаться в социальной навигации. Социальная навигация может применяться в прямой и косвенной форме. Прямая форма предполагает непосредственное общение и аннотирование контента, выраженное в комментариях пользователей, они могут оставить непосредственно на странице. Косвенная социальная навигация выражается в методах на основе истории и методах коллективной фильтрации. Навигация на основе истории визуализирует обобщенные или индивидуальные действия пользователей о текущей странице [8]. Так вариантами такого подхода является публикация количеству посещений страницы или времени, проведенного на ней другими пользователями и др. Подобная информация помогает понять, насколько данная страница релевантна сообществу пользователей. На основе визуализированных данных пользователь принимает решение о собственной навигации. Коллективная фильтрация - техника предоставления рекомендаций, основанных на предварительно выраженных мнениях или заинтересованности похожих пользователей. Коллективная фильтрация часто реализуется с помощью рейтингов или оценивания, а также индикаторов заинтересованности.

В связи с тем, что история адаптивных и интеллектуальных систем образования коренится в Интернет разработках, в них не уделяется достаточно широкое внимание моделированию коллективной работы участников учебного процесса. Однако этот аспект становится весьма важным в контексте современных образовательных Web-систем, учитывая тенденции развития Интернета.

Выводы. Обзор основных технологий адаптивных и интеллектуальных обучающих систем позволяет сделать выводы относительно особенностей применения таких технологий в контексте непрерывного образования. Одну из проблем многих систем можно выразить как принцип «полного интеллектуального руководства» - тенденции полного контроля над учебным процессом. Система, будучи педагогически сознательной, старается все сделать за пользователя и имеет тенденцию занимать позицию полной власти над учебным процессом. Целесообразным видится подход по принципу «интеллектуального партнерства», когда система, имея педагогическое сознание, направляет пользователя в духе советника и предоставляет ему широкие возможности для самостоятельной адаптации своего обучения. Это позволяет использовать «для помощи искусственному интеллекту» системы естественный интеллект ученика, что, несомненно, станет весомым вкладом в эффективность обучения и довольно сильно демократизирует образовательный процесс, соответствующий требованиям непрерывного обучения.

Кроме этого, учитывая анализ особенностей непрерывного образования, следует заметить, что рассмотренные адаптивные и интеллектуальные обучающие системы не уделяют достаточно непосредственного внимания некоторым специфическим требованиям

ДО: соответствия архитектуры системы явлению «информационного взрыва», профессиональной направленности обучения, междисциплинарности знаний.

Литература

1. Касьянов В.Н., Касьянова Е.В. // Методы и средства адаптивной гипермедиа. Вычислительные технологии. — 2004. — Т.9, Часть 2. — С. 333-341. 245
2. Peter Brusilovsky, Sergey Sosnovsky, Michael Yudelson, Amruth Kumar, Sharon Hsiao. // User Model Integration workshop. 5th International conference on Adaptive Hypermedia and Adaptive Web-Based Systems. — Hannover, 2008. P. 1—9.
3. Desmond Elliott, Frank Hopfgartner, Teerapong Leelanupab, Yashar Moshfeghi, Joemon M. Jose. // An Architecture for Life-long User Modelling. — 2009.
4. Loc Nguyen, Phung Do. // Learner Model in Adaptive Learning. — 2008.
5. Гагарін О.О., Титенко С.В. Дослідження і аналіз методів та моделей інтелектуальних систем безперервного навчання // Наукові вісті НТУУ "КПІ". — 2007. — № 6(56). — С. 37-48.
6. Федорук П. И. //Использование адаптивных и интеллектуальных технологий в системах дистанционного обучения УСиМ, № 5 - 2006.
7. Brusilovsky, P. and Millan, E. (2007) User models for adaptive hypermedia and adaptive educational systems. In: P. Brusilovsky, A. Kobsa and W. Neidl (eds.): The Adaptive Web: Methods and Strategies of Web Personalization. Lecture Notes in Computer Science, Vol. 4321, Berlin Heidelberg New York: Springer-Verlag, pp. 3-53.
8. Brusilovsky, P. and Henze, N. (2007) Open corpus adaptive educational hypermedia. In: P. Brusilovsky, A. Kobsa and W. Neidl (eds.): The Adaptive Web: Methods and Strategies of Web Personalization. Lecture Notes in Computer Science, Vol. 4321, Berlin Heidelberg New York: Springer-Verlag, pp. 671-696.
9. Conlan, O., O'Keeffe, I., Tallon, S.: Combining adaptive hypermedia techniques and ontology reasoning to produce dynamic personalized news services. In: Wade, V., Ashman, H., Smyth, B. (eds.) Proc. of 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2006). Lecture Notes in Computer Science, Vol. 4018. Springer Verlag (2006) 81- 90

УДК 612.014

КОМПЬЮТЕРНАЯ СИСТЕМА МОДЕЛИРОВАНИЯ ЖИЗНЕДЕЯТЕЛЬНОСТИ КЛЕТКИ КРОВИ

Фомичёв К. А., Герасимов И. Г.

Донецкий Национальный Технический Университет

Кафедра автоматизированных систем управления

E-mail: fka@bk.ru

Аннотация

Фомичёв К.А., Герасимов И.Г. Компьютерная система моделирования жизнедеятельности клетки крови. Обсуждается моделирование изменения геометрических параметров клеток в процессах их жизнедеятельности: прикрепление, фагоцитоз, трансформация, пролиферация, некроз, апоптоз. Разработана общая схема создания проекта и моделирования. Для моделирования используется профессиональная программная система «3Ds Max».

Общая постановка проблемы

Роль клеток в организме многообразна, прежде всего, по причине их специфичности в разных тканях и органах организма. Однако общими свойствами подавляющего большинства клеток являются такие процессы жизнедеятельности, как прикрепление, фагоцитоз, трансформация, пролиферация, некроз, апоптоз.

Наблюдение за клетками в процессах их жизнедеятельности – важнейший этап изучения их поведения. При этом наблюдать за процессами фагоцитоза, прикрепления, трансформации, пролиферации, некроза и апоптоза достаточно трудно, поскольку все клетки находятся в постоянном движении, их огромное количество, и обнаружить именно ту клетку, которая сейчас находится в определенном состоянии трудно. В реальном времени наблюдать определенный процесс жизнедеятельности клетки от его начала до конца затруднительно.

По таким причинам моделирование является удобным способом исследования жизнедеятельности клеток в реальном времени, для чего необходимо разработать соответствующую специализированную компьютерную систему.

Для выполнения поставленной задачи требуется 3D среда моделирования, а следовательно, компьютерный программный продукт, в котором возможно осуществить 3D моделирование объектов. Рассмотрено много вариантов такого программного обеспечения (Alias Maya 7.0, CyberMotion 3D Designer 10.0, Inivis AC3D 4.0.8, LightRay3D 1.3.4), но все же, самой оптимальной и удобной средой моделирования трехмерной графики была выбрана программа «3Ds Max», разработчиком которой является компания «Autodesk». «3Ds Max» предоставляет возможность с большей точностью изображать объекты, позволяет над этими объектами проводить различные действия, которые в последствии можно сохранить как видеоролик. Данный программный продукт выбран еще и потому, что включает в себя механизм расчета физики и позволяет моделировать поведение твердых и мягких тел с учетом силы тяжести и других воздействий [3]. Программа «3Ds Max» легкодоступна любому пользователю «Интернет», т.к. бесплатно распространяется на территории СНГ.

Анализ последних исследований и публикаций, по рассматриваемому вопросу

Последним исследованием в данной сфере является научная статья Герасимова И. Г., Привалова М. В., Гринченко А. А. «Прогнозирование параметров процесса прикрепления клеток крови *in vitro*» [1]. Разработана специализированная компьютерная система для прогнозирования параметров процесса прикрепления клеток крови на примере нейтрофилов на основании построения кинетической модели процесса. Снимки изображений нейтрофилов получали с помощью программы ASUS Live Version 4.6 B2. С помощью специально написанной программы проводили оконтуривание и расчет периметра нейтрофилов. Параметры кинетического уравнения процесса прикрепления, расчет среднего значения и доверительного интервала получали с использованием пакета статистической обработки R for Windows FAQ Version for R–2.4.1. Из нерешенных моментов данной работы следует выделить то, что прогнозирование осуществляется только для одного процесса – прикрепления, а другие процессы остаются не рассмотренными. Кроме того, программный продукт позволяет работать с уже готовыми снимками (изображениями) и не дает возможность моделирования процесса.

Цель исследования, постановка задачи

Целью исследования является создание программного продукта, который при определенных входных параметрах позволяет моделировать определенный (выбранный пользователем) процесс жизнедеятельности клетки. Результатом моделирования ожидается увидеть видеоролик (анимация) поведения клетки от начала процесса до его завершения.

Для того чтобы моделировать что-либо, необходимо иметь параметры и данные об объекте моделирования. Поэтому для данного исследования нам понадобятся геометрические параметры клетки крови (объем, площадь, радиус сечения), а также параметры межклеточного пространства (уровень pH, температура, ионный состав среды). Эти параметры необходимы для того, чтобы рассчитать скорость процессов жизнедеятельности клетки, а также для более точного реалистичного моделирования поведения клетки в клеточной среде.

Рассмотрим процессы жизнедеятельности клетки. На всех рисунках схематически показана клетка (замкнутый контур) с одним ядром (зачерненная область внутри клетки).

Прикрепление – процесс, в результате которого клетка прикрепляется к поверхности и плотно контактирует с ней и может расплываться. Схематически прикрепление изображено на рис. 1. Данный процесс является главным и первым в жизнедеятельности клетки, ибо только и исключительно после прикрепления клетка начинает развиваться, питаться, размножаться и т. д. [4]. Изначально клетка находится во взвеси над субстратом (рис. 1, а), далее она медленно движется к подложке и прикрепляется к ней (рис. 1, б). Далее клетка постепенно расплывается на подложке (рис. 1, в).



Рисунок 1 – Прикрепление клетки к поверхности. а) клетка над поверхностью; б) клетка прикрепилась к поверхности; в) клетка полностью расплалась по поверхности (конец прикрепления).

Фагоцитоз – процесс, в котором специально предназначенные клетки крови и тканей организма (фагоциты) захватывают и переваривают возбудителей инфекционных заболеваний и отмершие клетки [4]. Схематически фагоцитоз изображен на рис. 2. Изначально клетка уже прикреплена к плоскости, ибо выполнение любых процессов жизни клетки возможны только после ее прикрепления к субстрату (рис. 2, а). Клетка (фагоцит) поглощает бактерию: мембрана клетки обволакивает бактерию и тем самым бактерия оказывается внутри клетки (рис. 2, б). Далее уже происходит расщепление захваченного объекта. (рис. 2, в). Таким образом, клетка питается.

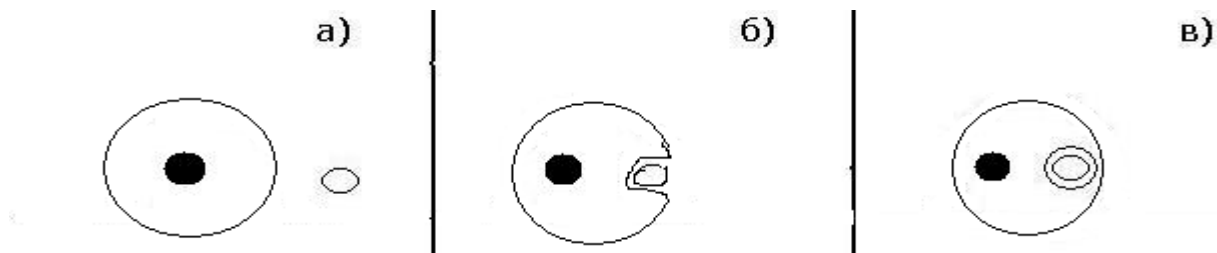


Рисунок 2 – Схема фагоцитоза. а) фагоцит и мелкая бактерия; б) клетка обволакивает мелкий объект; в) замыкание мембраны, погружение объекта с частью мембраны фагоцита внутрь клетки.

Пролиферация – размножение клеток (образование новых клеток) [4]. Схематически пролиферация изображена на рис. 3. В клетке происходит удвоение ДНК ядра (рис. 3, б). В ней образуется поперечная перегородка (рис. 3, в). Затем дочерние клетки расходятся (рис. 3, г). Таким образом, клетка размножается.

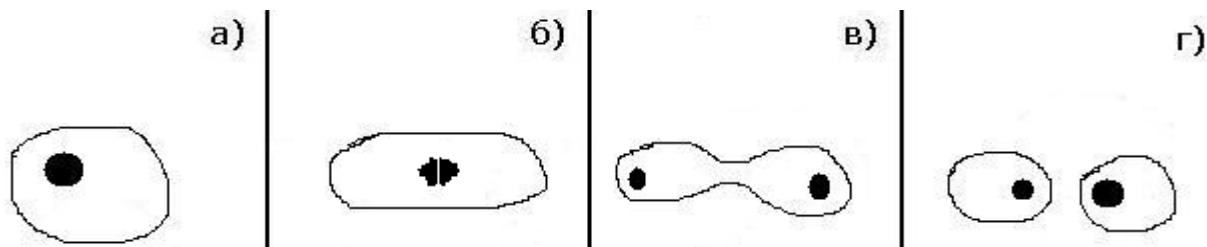


Рисунок 3 – Схема пролиферации клетки. а) исходный вид клетки; б) клетка увеличивается в размерах, удлинняясь; в) клетка постепенно делится на две клетки объединенных перегородкой; г) отделение клеток завершено.

Трансформация – изменение наследственных свойств клетки в результате мутации [4]. Схематически трансформация изображена на рис. 4. Клетка непредсказуемо меняет свои геометрические параметры (рис. 4, б, в).

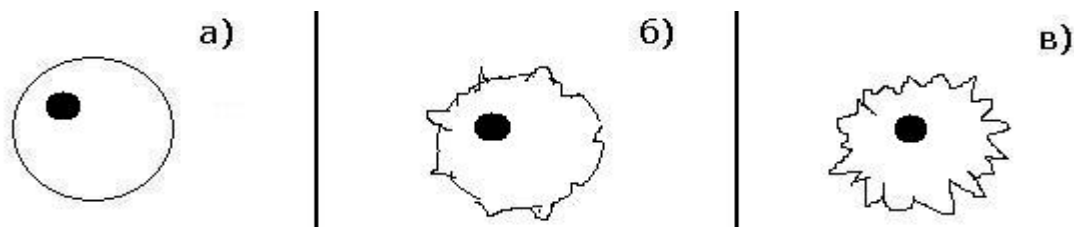


Рисунок 4 – Трансформация клетки. а) исходный вид клетки; б) первичная мутация, изменение клетки; в) конечный вид клетки в результате трансформации.

Некроз – это патологическая, форма клеточной гибели, в результате которой клетка набухает, увеличивается в размерах и разрушается [4]. Схематически некроз изображен на рис. 5. Геометрические параметры клетки увеличиваются настолько, что мембрана клетки разрывается и содержимое клетки выходит во внеклеточную сферу (рис. 5, в).

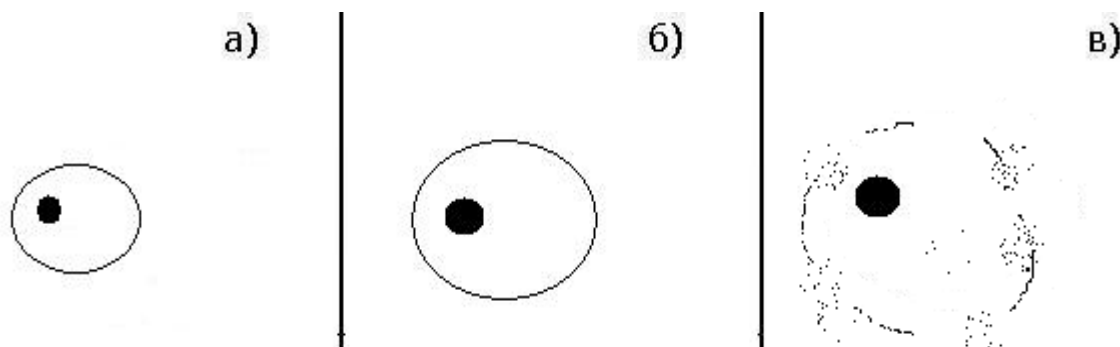


Рисунок 5 – Некроз клетки. а) исходный вид клетки; б) небольшое набухание клетки; в) сильное набухание клетки, приводящее к разрыву мембраны.

Апоптоз – форма гибели клетки, проявляющаяся в уменьшении её размера и распада на более мелкие части, которые поглощаются фагоцитами [4]. Схематически апоптоз изображен на рис. 6. Клетка уменьшается в объёме, фрагментирует и распадается на апоптотические тельца (рис 6, г).

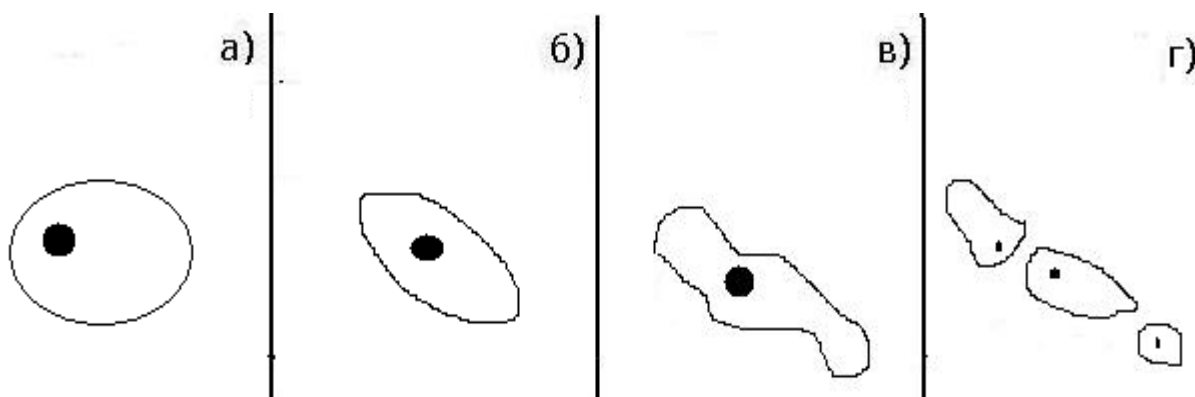


Рисунок 6 – Апоптоз клетки (вид сверху). а) исходный вид клетки; б) уменьшение размера клетки; в) постепенное деление клетки на части; г) конечный результат апоптоза.

Как известно, параметры межклеточного пространства влияют на поведение клетки. Влияние изменения некоторых факторов среды на протекание процессов жизнедеятельности клетки приведены в табл. 1. Таблица составлена по материалам статьи [2].

Таблица 1 – Влияние факторов внешней среды на скорость процессов жизнедеятельности клетки.

Процесс	Фактор									
	рН		Т		K ⁺		Na ⁺		Ca ²⁺	
	↑	↓	↑	↓	↑	↓	↑	↓	↑	↓
Прикрепление	-	-	-	-	?	?	?	?	?	?
Фагоцитоз	-	+	+	-	?	?	-	?	+	?
Пролиферация	-	+	+	-	?	?	?	?	-	?
Трансформация	-	+	+	-	?	?	?	?	?	?
Некроз	+	++	++	++	?	?	++	?	+	?
Апоптоз	+	+	++	-	+	+	+	+	+	-

Примечания: Увеличение «↑» и уменьшение «↓» влияющего фактора.

Обозначения: скорость процесса «+» – увеличивается; «++» – значительно увеличивает; «-» – уменьшается. Знак «?» указывает на то, что влияние данного фактора на процесс в литературе нами не обнаружено.

Выводы

Разработана схема специализированной компьютерной системы для моделирования процессов прикрепления, фагоцитоза, трансформацию, пролиферации, апоптоза и некроза клетки и отображения данных процессов жизнедеятельности клетки постепенно от начала до конца. Моделирование позволяет указывать исходные геометрические параметров клетки, а также параметры внешней среды, которая влияет на скорость процесса.

Литература

1. Герасимов И. Г., Привалов М. В., Гринченко А. А. Прогнозирование параметров процесса прикрепления клеток крови *in vitro*, // Наукові праці Донецького національного технічного університету. Сер. Обчислювальна техніка та автоматизація, 2008, - С. 1.
2. Герасимов И.Г. Влияние рН на деление клеток, их трансформацию и гибель // Вісник Донецького національного університету. Сер. А: Природничі науки, 2010, вип. 1.- С. 5.
3. Верстак В.А. Секреты мастерства. Учебник по 3D Max. Спб.: ЗАО Издатдом Питер, 2006. – 466 с.
4. Кузнецов С. Л., Пугачев М. К. Лекции по гистологии, цитологии и эмбриологии. – М.: Мед. информ. агентство, 2004. – 432 с.

УДК 371.263

ОЦЕНКА ЭФФЕКТИВНОСТИ КОНТРОЛЯ ЗНАНИЙ В ДИСТАНЦИОННЫХ СИСТЕМАХ ОБУЧЕНИЯ**Хохлова Е.О., Голубева Л.Г.**

Донецкий национальный технический университет

Факультет компьютерных наук и технологий

Кафедра компьютерных систем мониторинга

lesic.khokh@gmail.com**Аннотация**

Хохлова Е.О., Голубева Л.Г. Оценка эффективности контроля знаний в дистанционных системах обучения. Рассмотрены основные методы и модели оценки знаний, их параметры. Классификация типов тестовых заданий для дистанционных систем обучения. Проведен анализ методов оценки знаний для построения оптимизированной системы оценки.

Общая постановка проблемы:

Использование IT - технологий и дистанционного обучения открывает новые границы для непрерывного повышения квалификации учителя, а также поиска эффективных форм его взаимодействия со студентами. Эволюция методов обучения влечёт за собой совершенствование методов контроля знаний обучаемых.

Проверка и оценка знаний, умений и навыков студентов является важным структурным компонентом процесса обучения и в соответствии с принципом систематичности, последовательности и прочности обучения осуществляется в течение всего учебного года. Этим обуславливаются различные виды проверки и оценки знаний.

На современном этапе при оценке знаний обучаемых используется такая форма контроля знаний, как тестирование. Тесты – наиболее качественный и объективный способ оценивания обучаемых, позволяющий выявить степень овладения ими конкретными знаниями, умениями, навыками, а также позволяющий соотнести уровень достижений обучаемых по отдельному предмету в целом и по его отдельным разделам.

При проверке и оценке качества успеваемости важно выявлять, как решаются основные задачи обучения, т.е. в какой мере студенты овладевают знаниями, умениями и навыками. Существенное значение имеет также отношение студента к обучению и другие факторы, влияющие на успешность учебного процесса. Все это обуславливает необходимость применения различных методов проверки и оценки знаний.

Во многих отечественных и зарубежных вузах используют различные методы и программно-технические средства контроля знаний обучаемых [1]. Тем не менее, они не могут гарантировать получение достоверных данных не только об уровне знаний, умений и навыков обучаемых. Они, как правило, не ориентированы на выявление индивидуальных возможностей и способностей испытуемых [2].

Работы по гармонизации компьютерных технологий с традиционными формами обучения и контроля знаний обучаемых направлены на совершенствование дистанционного образования. Несмотря на имеющиеся достижения в этой области экзамены и зачёты по любым учебным дисциплинам в основном принимаются традиционным способом, т.е. после непосредственного общения обучающихся с обучаемыми. Сдача студентами зачётов и экзаменов преподавателям в виде тестов, проводимых с помощью компьютеров или других средств в местах близких к проживанию обучающихся по дистанционной форме обучения, все-таки требует их присутствия в установленное время в определенных аудиториях [3].

Исследования

В системе дистанционного образования используются практически все возможные организационные формы контроля, дополненные специально разработанными компьютерными программами, позволяющими снять часть нагрузки с преподавателя и усилить эффективность и своевременность контроля. Таким образом, применение новых образовательных технологий расширяет возможности контроля учебного процесса.

Основной задачей системы тестирования в дистанционных системах обучения является объективная оценка знаний тестирующего, выявление его слабых и сильных сторон, оценка результатов с указаниями по повышению результатов тестирования (в случае неудовлетворительного результата) [4].

Рассматривая предлагаемые в настоящее время методы компьютерного контроля знания и учитывая современные требования индивидуализации процесса обучения, отметим, что важным фактором, определяющим достоинство и преимущество того или иного метода контроля знаний, является степень (уровень) адаптации к индивидуальным характеристикам студентов и возможность применения оптимальных параметров проведения контроля знаний [5].

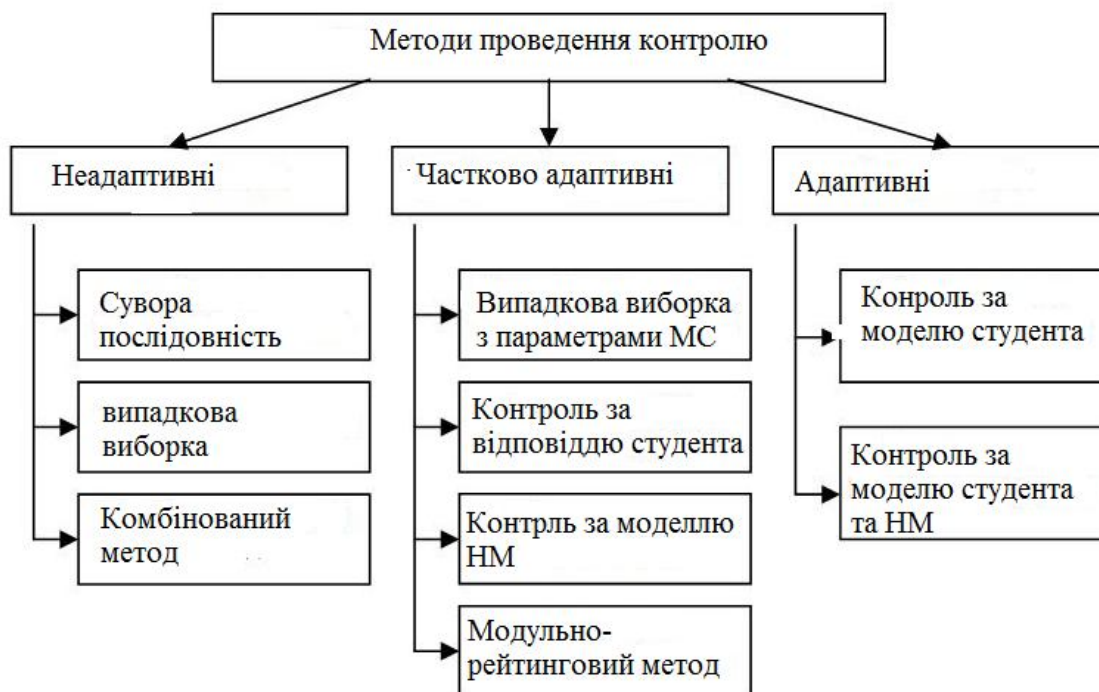


Рисунок 1 – Методы проведения контроля

Методы организации контроля знаний можно разделить на три класса:

- неадаптивные методы;
- частично адаптивные методы;
- полностью адаптивные методы.

Проанализировав каждый из методов контроля знаний, учитывая влияния параметров контроля знаний каждого из метода и учитывая свойства и требования к новой оптимизированной системе контроля знаний, сгенерирована последовательность построения модели оценки знаний [6].

Определение и оценка знаний представляет собой задачу распознавания, основанную на обучении. Решение проблемы оценки состоит из трех этапов:

- определение параметров контроля (обучение), которое выполняется до начала КЗ;

- сбор, анализ и преобразование данных, получаемых в процессе контроля (распознавание);
- выставление оценки за контрольную работу по завершении контроля (распознавания).

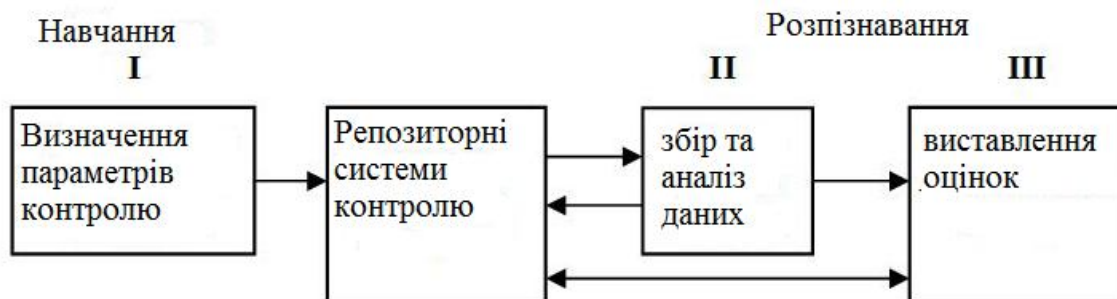


Рисунок 2 – Модель оценивания знаний при контроле

Для оценки знаний студентов применяются различные модели и алгоритмы, начиная с простейших, учитывающих только процент правильно выполненных заданий при двухбалльной системе оценки отдельного вопроса, и заканчивая сложными составляющими, в которых используются разнообразные параметры контроля и многобалльной системы оценки как отдельных задач, так и работы в целом. Все методы оценки предусматривают в процессе контроля знаний сбор данных о ходе контроля. Метод линейно-кусочной аппроксимации и модели на основе вероятностных критериев предполагают также вычисление некоторых функций, которые обычно используются для определения дальнейшего хода контроля.

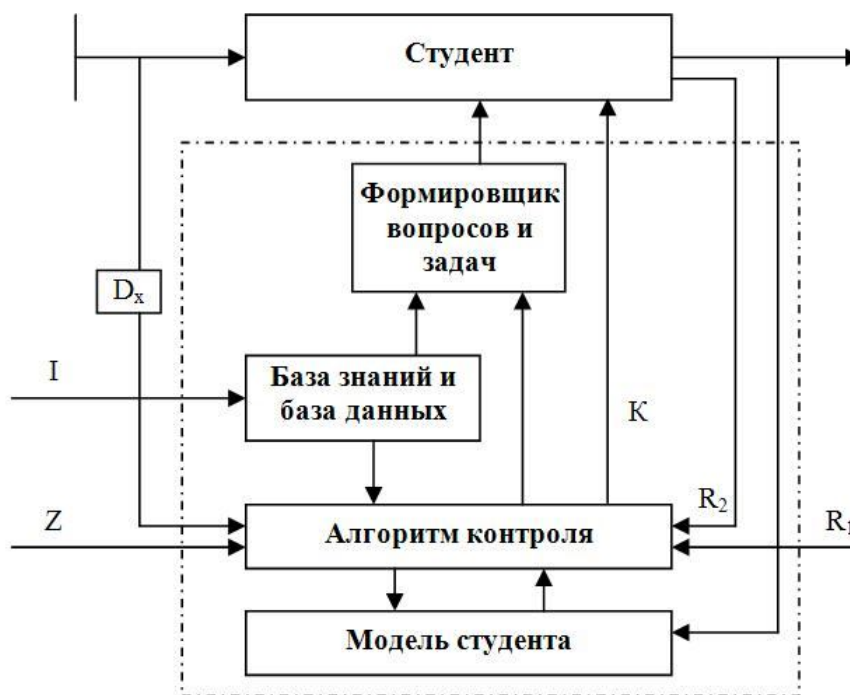


Рисунок 3 – Процесс управления адаптивным контролем знаний

Блок "Алгоритм контроля" выполняет следующие функции:

- анализ деятельности студента (проверка правильности его ответов и выполняемых

действий);

- управление процессом контроля знаний на основе выбранного метода;
- определение результатов контроля, которое обычно сводится к выставлению оценки студенту.

База знаний (БЗ) содержит методы и модели процесса контроля, а также совокупность знаний предметной области. База данных (БД) включает наборы вопросов и задач, предназначенных для проверки знаний студента и данные для формирования заданий. Контрольные задания могут генерироваться автоматически на основе БЗ [7].

Модель студента включает разнообразную информацию о студенте: предыстория обучения; результаты текущей работы (тип выполненных заданий, время выполнения заданий, число обращений за помощью и т.п.); личностные психологические характеристики (тип и направленность личности, репрезентативная система, способность к обучению, уровень беспокойства-тревоги, особенности памяти и др.); общий уровень подготовленности и другие.

Формировщик вопросов и задач используется для формирования и выдачи студенту очередного задания (вопросы или задания).

Контроль знаний осуществляется следующим образом: студент выполняет предложенное задание, и результат его работы помещается в модель студента. Блок "Алгоритм контроля" на основе анализа ответа студента, целей контроля и используемого метода проведения контроля, учитывая внешние ресурсы (например, возможности системы контроля) и внутренние ресурсы студента, а также состояние среды, определяет параметры задачи, которое должно быть предложено студенту. Формировщик вопросов и задач, получив от "Алгоритма контроля" данные о параметрах следующего задания, выбирает из БД и БЗ необходимую информации, формирует текст задания и выдает его студенту. В простейшем случае работа этого блока сводится к выбору нужного вопроса или задания из базы данных. После ответа на ряд сформированных вопросов [8], происходит оценка результатов тестирования по критериям (параметрам знаний).

Параметры контроля могут быть условно разделены на три группы: параметры, характеризующие отдельное задание и его выполнение, параметры, характеризующие работу ученика с набором контрольных заданий (количество заданий, количество ответов, количество правильных ответов, количество неправильных ответов; общее время, потраченное на выполнения задач, количество заданий, выполненных с превышением времени, количество обращений к справочной информации при выполнении заданий, уровень подготовки ученика, количество невыполненных задач, сложность, значимость и трудность контрольной работы и др.), параметры, используемые для настройки алгоритма, обычно задаются преподавателем, но могут иметь и заранее установленные значения (максимальный балл или оценка; предельные значения для выставления оценки и др.). Оценив влияние каждого из параметров контроля знаний на результат тестирования, создана система критериев. Система сравнивает полученный после блока тестирования результат с системой критериев и перенаправляет студента на нужный уровень (в зависимости от результата – предлагает пройти тест заново, перейти на средний, сложный, высокий уровень и т.д.) [9]. В зависимости от направления формируется новый блок вопросов, по которым (после прохождения блока) проводится снова анализ, оценка по параметрам знаний. Результаты анализируются, выводится результат и т.д. Все данные результатов тестирования фиксируются.

Выводы

В современные адаптивные системы обучения и контроля знаний следует включать несколько различных методов и моделей, чтобы преподаватель имел возможность выбрать метод проведения контроля и модель выставления оценки, отвечающие целям контроля и

наиболее подходящие для отдельного или группы студентов.

Эксперимент подтвердил положение о возможности интеграции в проектируемой и реализуемой системе контроля различных научных направлений (кибернетики, педагогики, психологии, теории управления качеством и др.) на основе широкого использования новых информационных технологий. Применение в программах контроля таких форм заданий, как множественный выбор, упорядочение, подстановка, трансформация и других, предусматривающих самостоятельность и творчество учащихся, проявление и диагностику различных компетентностей, может и должно сыграть важную роль в совершенствовании учебного процесса.

Установлено, что в условиях автоматизации процедур контроля появляется возможность не только унифицировать критерии оценки, но и присваивать «вес» каждому контрольному заданию и каждой допускаемой ошибке, влияющей на интегральную оценку знаний, что позволяет решить проблему, связанную с преодолением субъективизма в оценке знаний.

Таким образом, в исследовании решена актуальная научная задача, заключающаяся в разработке методологии и технологии контроля знаний, интегрированного в систему управления качеством образования.

Полученные в работе результаты предполагается использовать для совершенствования систем управления качеством образования, повышения обоснованности принимаемых в образовательной сфере решений.

Литература:

1. О внедрении новых методик оценивания знаний студентов - <http://www.culturalnet.ru/main/getfile/152> (5.11.10)
2. Пивоварова М.А. Информационные технологии в учебном процессе: возможности и опасности // Социально-гуманитарные знания.-2000.-№5.-С.129-140.
3. Разработка заданий для объективной оценки знаний студентов - <http://www.science-education.ru/33-1229> (05.12.10)
4. Галаган А.И. Проблемы интернационализации и контроля качества высшего образования и подходы к их решению в ряде зарубежных стран // Социально-гуманитарные знания.-2002.-№3.-С.221-228.
5. Карташова О.В. Использование адаптивной системы тестирования АСТ-тест для контроля знаний при дистанционном изучении темы «Базы данных» // Материалы международной конференции-выставки «Информационные технологии в образовании ». - Москва, 2001.
6. Попов Д.И. Подсистема адаптивного тестирования среды дистанционного обучения // Материалы международной конференции-выставки «Информационные технологии в образовании». - Москва, 2002.
7. Рынок систем дистанционного образования - <http://www.distance-learning.ru/db/el/9718E35D3B7363BCC3256E1F005D26F8/doc.html> (10.01.11)
8. Матушанский.Г.У. Проектирование педагогических тестов для контроля знаний// Информатика и образование- М.: 2000-№6.- стр.7-10.
9. Белоус Н.В., Войтович И.В. Автоматизированная система оценивания тестовых заданий разных форм // Вестник ХНТУ. – 2006. – № 1 (24). – С. 422-426.

УДК 3054

МЕТОД ВИРІШЕННЯ ДВОКРИТЕРІАЛЬНОЇ ТРАНСПОРТНОЇ ЗАДАЧІ ТА ЙОГО ОЦІНКА

Шевченко В.С.

Національний технічний університет України «Київський політехнічний інститут»
кафедра автоматизованих систем обробки інформації та управління
E-mail: akyлина90@gmail.com

Анотація

Шевченко В.С. Метод вирішення двокритеріальної транспортної задачі та його оцінка. В статті розглянута двокритеріальна транспортна задача за критеріями часу та вартості. Запропоновано евристичний метод вирішення двокритеріальних задач та проведені дослідження по оцінці його ефективності.

Вступ

Багато практичних проблемних ситуацій, які пов'язані з доставкою, транспортуванням та перевезенням продукції можуть бути зведені до транспортної задачі. В класичних постановках цієї задачі використовується один з критеріїв: сумарні транспортні витрати або загальний час перевезення усієї продукції [1,2]. Часто виникає ситуація, коли кінцевий користувач хоче отримати оптимальний (або близький до оптимального) розв'язок за цими двома критеріями.

Постановка задачі

Нехай існує m пунктів виробництва $A_i, i = \overline{1, m}$ з обсягами пропозиції $a_i, i = \overline{1, m}$ одиниць продукції та n пунктів споживання цієї продукції $B_j, j = \overline{1, n}$ з величинами попиту $b_j, j = \overline{1, n}$ одиниць продукції. Для кожної пари $A_i - B_j, (i = \overline{1, m}, j = \overline{1, n})$ відомі вартість перевезення одиниці продукції c_{ij} та витрати часу t_{ij} на здійснення перевезень (і допускається, що останні не залежать від обсягів перевезень $x_{ij}, i = \overline{1, m}, j = \overline{1, n}$). Потрібно встановити такі обсяги перевезень від кожного виробника до кожного споживача, щоб час T , який витрачається на перевезення та сумарні витрати на перевезення були б мінімальними за умови, що потреби всіх споживачів будуть задоволені.

Математична модель задачі

Критерій, за яким оцінюються час на перевезення:

$$\min T = \max_{x_{ij} > 0} \{t_{ij}\} \quad (1)$$

Критерій, за яким оцінюються сумарні витрати [2]:

$$\min z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (2)$$

Обмеження:

$$\sum_{j=1}^n x_{ij} = a_i, i = \overline{1, m} \quad (3)$$

$$\sum_{i=1}^m x_{ij} = b_j, j = \overline{1, n} \quad (4)$$

$$x_{ij} \geq 0, i = \overline{1, m}, j = \overline{1, n} \quad (5)$$

Розв'язання задачі

Запропоновано такий алгоритм розв'язання.

Етап 1. Розв'язання однокритеріальної ТЗЛП за критерієм вартості (без врахування часу на перевезення) методом потенціалів [2]. Результатом розв'язання є план перевезень x^c з відповідними мінімальними витратами $C(x^c)$ та загальним часом перевезень $T(x^c)$.

Етап 2. Розв'язання однокритеріальної задачі за критерієм часу не зважаючи на критерій вартості [1]. Результатом розв'язання задачі є план перевезень x^t , якому відповідає загальний мінімальний (або близький до мінімального) час перевезень $T(x^t)$ та відповідні витрати $C(x^t)$.

Етап 3. Об'єднання розв'язків попередніх етапів у кінцевий результат.

На цьому етапі виконується комбінування розв'язків x^t та x^c за наступним алгоритмом:

3.1. Якщо $T(x^c) \leq T(x^t)$, то розв'язком початкової двокритеріальної задачі (1) - (5) є розв'язок x^c . Кінець роботи алгоритму.

В такому випадку отримаємо мінімальні витрати за мінімальний час. Зазначимо, що такий результат є ідеальним (навіть утопічним) і досягнути його майже не можливо, оскільки очевидно, що у більшості випадків $T(x^c)$ буде набагато більшим за $T(x^t)$.

3.2. Якщо $\forall (i, j), i = \overline{1, m}, j = \overline{1, n}: x_{ij}^c = x_{ij}^t$, то $x_{ij}^H = x_{ij}^t = x_{ij}^c$, де x_{ij}^H - новий план перевезення (розв'язок двокритеріальної задачі (1)-(5)). Якщо $\forall (i, j), i = \overline{1, m}, j = \overline{1, n}: x_{ij}^c \neq x_{ij}^t$, то $x_{ij}^H = x_{ij}^t$. Тобто якщо плани x^t та x^c співпадають повністю або взагалі не співпадають (немає жодного спільного шляху), то розв'язком задачі є план x^t . Кінець роботи алгоритму.

3.3. Знаходження перетину двох планів x^t та x^c : якщо $x_{ij}^c = x_{ij}^t$, то $x_{ij}^H = x_{ij}^t = x_{ij}^c$, $i = \overline{1, m}, j = \overline{1, n}$. В результаті отримаємо x^H - частковий розв'язок, який доповнюється до допустимого на наступних кроках алгоритму.

4. Доповнення часткового розв'язку x^H до допустимого.

4.1. Перерахування попиту та пропозиції для усіх перевезень, що увійшли у розв'язок x^H :

$$\begin{aligned} a_i^H &= a_i - x_{ij}^H, \\ b_j^H &= b_j - x_{ij}^H, \end{aligned}$$

де a_i^H, b_j^H - нові (змінені) пропозиція та попит відповідно.

4.2. Визначення величини ΔT - величини, на яку можна збільшити загальний час перевезень.

4.3. Визначення максимально допустимого часу перевезень $T_{\max} = T(x^t) + \Delta T$.

4.4. Для кожного з пунктів призначення $B_j, j = \overline{1, m}$, для яких сумарний обсяг поставок на поточний момент дорівнює нулю виконати:

4.4.1. Розрахунок часу та витрат на перевезення продукції з розв'язку x^t . Позначимо їх відповідно t_j^1 та c_j^1 .

4.4.2. Розрахунок часу та витрат на перевезення продукції з розв'язку x^c . Позначимо їх відповідно t_j^2 та c_j^2 .

4.4.3. Якщо $t_j^2 \leq T_{\max}$, то перейти на пункт 4.4.3.1, інакше на пункт 4.4.3.2.

4.4.3.1 Якщо $c_j^2 \leq c_j^1$, то j -ий стовпець з розв'язку x^c копіюється у j -ий стовпець розв'язку x^t . Перейти на пункт 4.4.

4.4.3.2 j -ий стовпець з розв'язку x^t копіюється у j -ий стовпець розв'язку x^c .

4.5. Обрати пункти призначення, у яких продукція ще не була доставлена повністю та доставити в них залишкову у пунктах виробництва продукцію. В результаті отримаємо новий план перевезень x^H з відповідними витратами $C(x^H)$ та загальним часом перевезень $T(x^H)$.

5. Якщо $C(x^H) \geq C(x^t)$, то вважаємо, що розв'язком задачі (1)-(5) є розв'язок x^t . Кінець алгоритму.

Висновки

Отже, отримали розв'язок x^H , де час на доставку товару більше ніж $T(x^t)$ та сумарні транспортні витрати більші за $C(x^c)$. При заданому користувачем допустимому збільшенню часу $T(x^t)$ на доставку товару, ми досягли зменшення сумарних транспортних витрат $C(x^t)$. Такий розв'язок є проміжним між розв'язками, які були отримані при розв'язанні задач першого та другого етапів.

Експериментальним способом було встановлено, що якщо на початку роботи алгоритму задати незмінне на час роботи алгоритму ΔT (тобто одне й те саме число використовувати на кожній новій ітерації алгоритму), то новий план перевезень, отриманий в кінці роботи алгоритму, співпадає з одним з планів перевезення продукції x^t або x^c . Тобто використовувати запропонований алгоритм стає недоцільним.

Основною проблемою даного алгоритму є відсутність формального правила визначення нового ΔT на кожній ітерації. Саме тому пропонується на кожній новій ітерації алгоритму генерувати нове значення ΔT (наприклад, в певних заданих на початку роботи алгоритму межах, які обґрунтовані користувачем).

Було також встановлено, що у випадку коли витрати на доставку прямопропорційні часу на перевезення товару, то у більшості випадків (87%) розв'язок ТЗЛП за критерієм часу співпадає з розв'язком ТЗЛП за критерієм вартості. У такому разі маємо ідеальний розв'язок, який пропонує такий план перевезень, при якому час та витрати на доставку є мінімальними, тобто ідеальний розв'язок.

ЛІТЕРАТУРА

1. Селезньова О.О, Шевченко В.С. Дослідження двокритеріальної транспортної задачі: матеріали III всеукраїнської науко – практичної конференції «Інформаційні технології та автоматизація - 2010», Одеса, 14 – 15 жовтня, 2010 г.: збірник доповідей, 2010. – С.49-50.

2. Таха Х.А. Введение в исследование операций [Текст]/ Хемди А. Таха// М.: «Вильямс», 2007. – [7е изд.]. – 912 с.

3. Транспортна задача лінійного програмування: Методичні вказівки до самостійної роботи та практичних занять з дисципліни «Математичні методи дослідження операцій» для студентів спеціальності «Інформаційні управляючі системи та технології»/Укл.:О.Г.Жданова, С.С.Жевновак, В.М.Кузнецов – К.: НТУУ «КПІ», 2000. – 60с.

4. Калихман И. Л. Линейная алгебра и программирование. –М. Высш. Шк. 1967. – 428с.

УДК 004.94

О КРИТЕРИЯХ АНАЛИЗА ПРОЦЕССОВ КОМПОЗИЦИИ ВЕБ-СЕРВИСОВ**Шкарупило В.В., Кудерметов Р.К.**

Запорожский национальный технический университет, г. Запорожье

кафедра компьютерных систем и сетей

E-mail: vadshkar@yandex.ru**Аннотация**

Шкарупило В.В., Кудерметов Р.К. О критериях анализа процессов композиции веб-сервисов. Предложены критерии анализа процессов композиции веб-сервисов. На основе предложенных критериев разработана организация процесса автоматизированной композиции веб-сервисов. Организация включает спецификацию, верификацию, валидацию, планирование и моделирование композитных веб-сервисов.

Общая постановка проблемы

Специфика композиции веб-сервисов связана с основными свойствами парадигмы сервис-ориентированной архитектуры (SOA), такими как гибкость, интероперабельность, слабая связанность, динамичность изменения функциональных и нефункциональных (QoS, Quality of Services) требований к веб-сервисам. Практическое значение имеет возможность автоматизации композиции, особенно, при рассмотрении ее как процесса «end-to-end» [1], который предполагает полный цикл создания композитного сервиса – от поступления запроса на услугу со стороны клиента до получения им результатов услуги, предоставленной композитным веб-сервисом. Этот процесс может включать множество типов активностей и их этапов, номенклатура и последовательность которых пока еще недостаточно освещена в публикациях и не имеет общепринятых спецификаций и/или стандартов. В данной работе анализируются возможные критерии для оценки путей построения автоматизированной композиции веб-сервисов.

Постановка задачи

Процесс автоматизации композиции веб-сервисов можно организовать различными путями. Например, используя первые подходящие и/или известные атомарные (не композитные) веб-сервисы, применяя оркестровку и хореографию к веб-сервисам, спецификации которых найдены в корпоративных и публичных реестрах UDDI, ebXML, по результатам имитационного моделирования комбинаций и вариантов различных атомарных веб-сервисов, исследуя их семантические модели. Однако для оценки этих путей необходимо применять критерии, с помощью которых можно определить эффективность процесса композиции и качество получаемых композитных сервисов. Поэтому актуальным является исследование и обоснование таких критериев, а также поиск путей автоматизированной композиции сервисов, удовлетворяющих выбранным критериям.

Критерии анализа процесса композиции веб-сервисов

Рассматривая процесс композиции сервисов, следует принимать во внимание основные артефакты системного инжиниринга сложных систем, такие как его фазы, дисциплины, распределенность разработки, возможные различные понятийные аппараты у провайдеров и, соответственно, семантические контексты у сервисов и т. п. Поэтому нами, прежде всего, выделены такие свойства процесса автоматизации композиции веб-сервисов:

- спецификация требований клиента и свойств веб-сервисов;
- верификация и валидация (V&V) композитных веб-сервисов;
- планирование процесса композиции.

Для спецификации веб-сервисов и их композиций можно применять графические, текстовые, формальные и др. способы. В табл. 1 приведен их краткий анализ.

Таблица 1 – Характеристики способов спецификации композиции

Способ спецификации	Достоинства	Средства спецификации
Графическая	Наглядность, возможность представления в виде XML-схем	IDEF0, UML, SysML
Текстовая	Стандартизация	WS-BPEL, WS-CDL
Формальная	Возможность автоматизированных процессов V&V	Темпоральные логики (LTL, TLA)

Наибольший интерес для автоматизированной композиции представляют машинно-ориентированные способы спецификации. В настоящее время развиваются способы и инструментарии автоматизированной генерации спецификаций, программного кода из спецификаций и их обратного преобразования (Rational, StarUML, NetBeans, Altova, Protege, плагины для Eclipse). При этом наиболее перспективным считается использование диаграмм языка UML и производных от него языков (например, SysML, SoaML), диалекты языка XML (WSDL, gWSDL, tModel для UDDI, OWL-S, SOAP).

На наш взгляд наиболее важными являются следующие критерии оценки процесса автоматизированной композиции веб-сервисов:

- 1) способ спецификации (графическая, текстовая, формальная);
- 2) полнота обеспечения V&V композитных сервисов;
- 3) учет нефункциональных (QoS) требований;
- 4) практическая применимость;
- 5) возможность планирования процесса композиции.

Способ спецификации – дает возможность определить, предполагает ли подход формальную спецификацию композиции и, как следствие, автоматизации процессов V&V. Возможность обеспечения V&V формальной спецификации необходима для проверки корректности композиции и ее характеристик. Важно провести разграничение между понятиями верификации и валидации. Под верификацией понимают проверку параметров формальной спецификации, а под валидацией – проверку корректности формальной спецификации [2].

Учет QoS свойств композиции позволяет повысить качество веб-сервисов, например, в качестве параметра QoS может служить время отклика экземпляра веб-сервиса [3].

Практическая применимость определяет эффективность подхода в реальных условиях (в случае участия в процессе композиции сотен экземпляров веб-сервисов).

Процесс композиции веб-сервисов

Для иллюстрации предложенных критериев рассмотрим один из возможных путей организации автоматизированной композиции веб-сервисов (рис. 1). Суть этой организации заключается в последовательном выполнении следующих шагов:

- 1) графическая спецификация композиции с использованием языка UML;
- 2) формальная спецификация композиции с применением TLA;
- 3) верификация формальной спецификации при помощи инструментария TLA Model Checker;
- 4) валидация модели композиции в моделирующей среде DEVS Suite;
- 5) планирование композиции (с оптимизацией по параметру QoS);
- 6) имитационное моделирование композиции на основе агентного подхода.

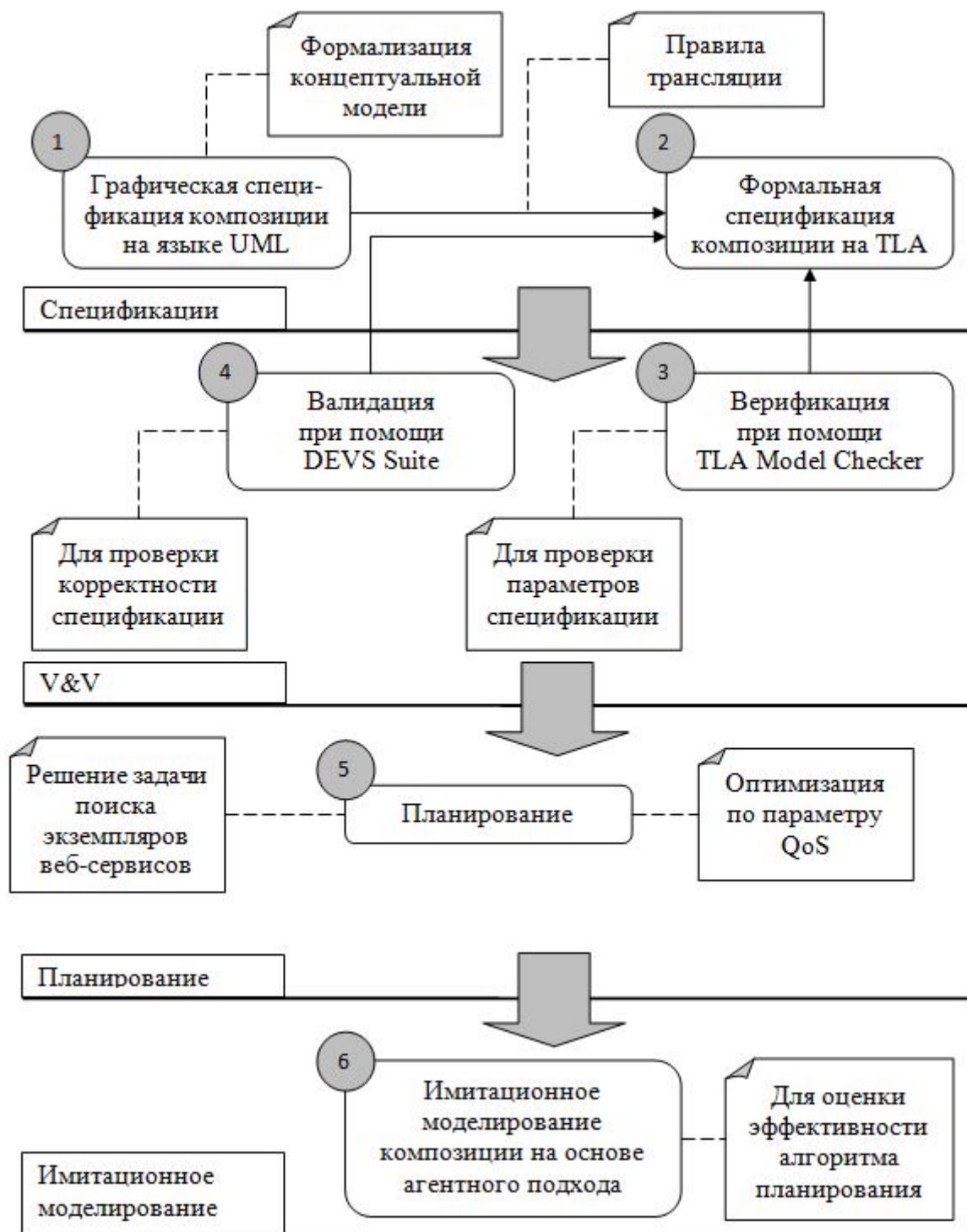


Рисунок 1 – Процесс композиции веб-сервиса

Начальным шагом является графическая спецификация концептуальной модели композитного веб-сервиса на языке UML, с целью определения и формального отображения архитектуры веб-сервиса, т.е. основных его элементов и связей между ними. Формальное графическое представление архитектуры повышает вероятность успешной валидации модели, построенной на основе формальной спецификации композиции.

Рассматривая композицию как процесс «end-to-end», важно выработать набор правил трансляции из графической спецификации в формальную. В большинстве подходов, предполагающих формальную спецификацию композиции (шаг 2), в качестве языка

спецификации используют темпоральную логику LTL (Linear Temporal Logic). Для нашего процесса мы предполагаем в качестве логики спецификации использовать TLA (Temporal Logic of Actions). TLA, на наш взгляд, имеет более полный набор синтаксических средств описания взаимодействия параллельных и распределенных процессов, что является критическим для формального описания композиции веб-сервисов. Другим преимуществом использования TLA является возможность формальной верификации. В случае логики LTL для формальной верификации необходимо создание отдельной спецификации верифицируемых параметров в пакете CADP (Construction and Analysis of Distributed Processes). Эта специфика требует дополнительных усилий и повышает вероятность ошибок.

При использовании TLA спецификация и верификация могут быть выполнены с помощью инструментария TLA Toolbox, который имеет специальное средство для верификации спецификаций TLA Model Checker [4]. Использование последнего предполагается на третьем шаге – формальной верификации композитного веб-сервиса.

Четвертый шаг – валидация формальной спецификации веб-сервиса путем создания дискретно-событийной имитационной модели и моделирование с помощью ее функционирования композитного веб-сервиса, например в DEVS Suite. Для этого необходимо разработать набор правил трансляции формальной спецификации в Java-код, описывающий DEVS-модели. Описание возможной обратной трансляции приведено в [5].

Пятый шаг процесса предполагает планирование композиции. Этот шаг также включает задачу поиска экземпляров веб-сервисов и предполагает оптимизацию по параметру QoS (время отклика экземпляра веб-сервиса).

В качестве заключительного шага выступает имитационное моделирование процесса композиции на основе агентного подхода, что обосновано введенным критерием практической применимости и наибольшим соответствием агентного подхода моделируемой предметной области.

Выводы

В работе обоснованы следующие критерии анализа подходов к композиции веб-сервисов: способ спецификации, полнота обеспечения V&V композитных сервисов, учет нефункциональных (QoS) требований, практическая применимость, возможность планирования процесса композиции. Продемонстрирована возможность их использования для предложенной организации автоматизированной композиции веб-сервисов.

Литература

1. Agarwal, V. Synthy: A System for End to End Composition of Web Services [text] / V. Agarwal, B. Srivastava // Journal of Web Semantics. – 2005. – Vol. 3, No. 6. – P. 311 – 339.
2. Foster, H. Using a Rigorous Approach for Engineering Web Service Compositions: A Case Study [text] / H. Foster, S. Uchitel, J. Magee, J. Kramer, M. Hu // Proceedings of the IEEE International Conference on Service Computing (Orlando, Florida, USA, July 11 – 15, 2005). – P. 217 – 224.
3. Bin, X. Efficient Composition of Semantic Web Services with End-to-End QoS Optimization [text] / X. Bin, L. Sen, Y. Yixin // Tsinghua Science and Technology. – 2010. – Vol. 15, No. 6. – P. 678 – 686.
4. Chaudhuri, K. The TLA+ Proof System: Building a Heterogeneous Verification Platform [text] / K. Chaudhuri, D. Doligez, L. Lamport, S. Merz // Proceedings of the 7th International colloquium conference on Theoretical aspects of computing (Rio Grande do Norte, Brazil, September 1 – 3, 2010). – p. 44.
5. Cristia, M. A TLA+ Encoding of DEVS Models [text] / M. Cristia // Proceedings of International Modeling and Simulation Multiconference (Buenos Aires, Argentina, February 8 – 12, 2007). – P. 17 – 22.

УДК 004.514

ИСПОЛЬЗОВАНИЕ ОСОБЕННОСТЕЙ ЦВЕТОВОСПРИЯТИЯ ПОЛЬЗОВАТЕЛЯ ПРИ РАЗРАБОТКЕ ИНТЕРФЕЙСА ПРОГРАММНОГО ПРОДУКТА**Якушева Е.Н.¹, Мирошкин А.Н.²**

Донецкий национальный технический университет,

¹ кафедра компьютерных систем мониторинга,² кафедра компьютерной инженерииE-mail: prekrasnius@gmail.com MiroshkinAN@gmail.com**Аннотация**

Якушева Е.Н., Мирошкин А.Н. Использование особенностей цветовосприятия пользователя при разработке интерфейса программного продукта. Обозначены основные недостатки существующих интерфейсов программных продуктов. Приведены примеры адаптированных программных средств. Представлена классификация нарушений цветовосприятия. Определен подход формирования требований к разработке универсальных интерфейсов. Определены дальнейшие цели и перспективы.

1. Общая постановка проблемы

Почти 10% всего населения планеты имеют нарушения цветовосприятия [1]. Показатель достаточно высок, и, ввиду этого, целесообразной является разработка объективных критериев выбора цветовой схемы оформления, простой для восприятия при дальтонизме.

Разработчики программного обеспечения реализуют интерфейс программ исходя из собственных вкусовых предпочтений. На сегодняшний день не существует единых объективных критериев оценки цветовой решения дизайна. Большинство рекомендаций по использованию цвета выдаются колористами для предотвращения негативного психофизиологического воздействия цвета на пользователя. Тем не менее, в условиях отсутствия универсальных требований к цветовой оформлению, колористический дизайн программы произвольно выбирается создателями, которые могут исходить только из личного опыта или некоторых уже устоявшихся традиций оформления. При этом нередко появляются непреодолимая трудность для восприятия программного продукта цветоаномалами – людьми с нарушениями цветовосприятия. Так, во всемирно известном текстовом редакторе Microsoft Word при автоматической проверке правописания разные типы ошибок подчеркиваются волнистыми линиями красного и зеленого цвета. Для многих дальтоников эти линии выглядят одинаково. Разработчики программы учли свою недоработку, и в новых версиях программы появилась возможность установки синего цвета волнистых линий. Следует отметить, что самостоятельно найти эту настройку является весьма проблематичной задачей для неопытного пользователя.

Ярким примером учета особенностей зрения различных людей является интерфейс операционной системы Microsoft Windows – его нормально воспринимают и большинство дальтоников, и люди с нормальным цветовым зрением, а специальные схемы повышенной контрастности позволяют использовать систему даже монохроматам – людям, не различающим цвета вообще.

Дизайн популярных социальных сетей «Facebook» и «ВКонтакте» выполнен преимущественно в синем цвете. Это обусловлено тем, что синий цвет одинаково хорошо различим большинством дальтоников и цветоаномалов. Однако предпочтение только синей гаммы не является универсальным методом, так как тританопы не различают оттенков синего.

2. Исследования

В сетчатой оболочке глаза находится два вида светочувствительных элементов: светочувствительные палочки и цветочувствительные колбочки [2]. Видимый спектр света воспринимается человеком с помощью колбочек. Существует три типа колбочек. Каждый из них обладает собственной спектральной чувствительностью. Первый тип воспринимает световые волны с длиной от 400 до 500 нм (синий), второй - от 500 до 600 нм (зеленый) и третий - от 600 до 800 нм (красный). Видение всех красок мира обеспечивается «складыванием» именно этих трех цветов в нашем мозге. Число рецепторов, отвечающих за восприятие определенных длин волн, у каждого человека индивидуально. Восприятие цветов изменяется с возрастом, зависит от остроты зрения и даже от национальности человека [3]. Люди с нормальным цветным зрением имеют в колбочках все три пигмента (красный, зелёный и синий) в необходимом количестве. Их называют трихроматами. В случае ослабления восприятия одного из цветов цветовое зрение может нарушаться. Примерно 10% людей страдают нарушениями цветового зрения – так называемым дальтонизмом. Дальтоники делятся на дихроматов, у которых ослаблено восприятие одного цвета, и монохроматов, различающих только градации серого. Монохроматизм встречается очень редко (приблизительно 1 из 2325 человек) [4]. Более всего распространен дихроматизм. Различают следующие виды дихроматизма.

Протанопия – слабость восприятия красного цвета. Дейтеранопия – зеленого. Тританопия – синего. У протанопов максимум спектральной чувствительности глаза сдвинут к 540 нм (лимонный). Максимум спектральной чувствительности дейтеранопы составляет 560 нм (желтый). И протанопам, и дейтеранопам средняя часть спектра (зеленый) представляется малонасыщенной и при значении световой волны ~495 нм – нейтрально-серой. Тританопам длинноволновый конец спектра (красно-оранжевый) представляется алым, а по мере приближения к нейтральной точке (~570 нм) цвета становятся всё более сероватыми. Со стороны коротких волн цветовой тон им представляется зелено-голубым с макс. насыщенностью при ~470 нм и резким её падением в конце спектра. Определенная часть спектра кажется цветоаномалам ахроматической. Для протанопы эта область между 480 и 495 нм (зелено-голубой), для дейтеранопы – между 495 и 500 нм (зеленый). Область спектра между 565 и 575 нм (ярко-желтый) тританопы воспринимают как ахроматический.

Пик чувствительности колбочек приходится на длину волны 564 нм для красного, 534 нм для зеленого и 420 нм — для синего цвета. В сумерках – 498 нм.

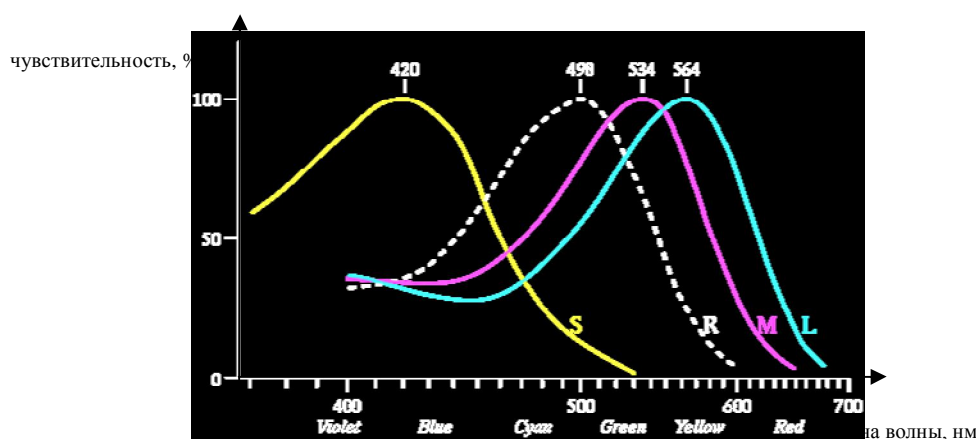


Рисунок 1 – График чувствительности глаза к различным цветам

Троичная система восприятия цвета широко используется в компьютерной индустрии. Всемирно известная цветовая модель RGB основана на естественном восприятии цвета человеком. Человеческое зрение реагирует отдельно на короткие, средние и длинные волны, которым соответствуют отдельные рецепторы сетчатки. Этим длинам волн соответствуют

Красный (Red), Зеленый (Green) и Синие (Blue) цвета, которые являются каналами этой цветовой модели. Программы, работающие с цветами, активно используют RGB режим, основанный на смешивании первичных цветов. Смешивая свет трех длин волн, и изменяя их интенсивность, можно получить любой существующий цвет. Смесь, состоящая из 100% каждого цвета, дает белый цвет. Отсутствие всех цветов дает черный цвет.

Основные спектральные цвета характеризуются следующей длиной волны: фиолетовый - 380-450 нм; синий - 450-480 нм; голубой - 480-510 нм; зеленый - 510-550 нм, желто-зеленый - 550-575 нм; желтый - 575-580 нм; оранжевый - 585-620 нм; красный - 620-770 нм [5]. Нелинейным преобразованием модели RGB является модель HSB — цветовая модель, в которой координатами цвета являются:

- Hue – цветовой тон;
- Saturation – насыщенность;
- Brightness – яркость.

3. Рекомендации

На основании рассмотренных значений, можно сделать заключение о наиболее предпочтительных принципах цветового оформления интерфейсов:

- следует избегать интенсивных, насыщенных спектральных цветов, а особенно их сочетаний (например, красного и синего) так как они могут вызвать болевые ощущения у пользователя;
- следует избегать контрастов между оттенками красного, коричневого, желтого и зеленого, так как большинство цветоаномалов не видит разницы между ними;
- следует избегать контрастов между оттенками фиолетового, синего и голубого цветов, так как большинство цветоаномалов не видит разницы между ними;
- большинство цветоаномалов не видит разницы между темно-фиолетовым, темно-коричневым и черным цветами;
- салатный и желтый для большинства дейтеранопов неразличимы;
- серый, розовый и голубой для большинства дейтеранопов неразличимы;
- восприятие желтого потока излучения с длиной волны 575 нм и синего с 475 нм будет одинаковым как для трихроматов, так и для протанопов и дейтеранопов;
- восприятие красного с 660 нм и сине-зеленого 485 нм будет одинаковым как для трихроматов, так и для тританопов.

Выводы

Представленный подход не универсален, хотя успешно применим при работе с текстом и изображениями. Следовательно, для уточнения закономерностей появляется необходимость в установлении границ цветового спектра и вычислении длин световых волн, не различимых при цветоаномалии.

Параметр Hue соответствует длине световой волны. Планируется установить зависимость переменной Hue от действительного спектрального значения цвета, для точной корректировки палитр программ.

Полученные зависимости можно успешно применять в области телевидения, компьютерной графики, мобильных технологий, печатных СМИ.

Существуют алгоритмы преобразования изображения из «трихроматного» в «дихроматное». В дальнейшем планируется найти обратный алгоритм для усиления контраста между неразличимыми цветами, чтобы сделать простым и доступным распознавание оттенков для каждого пользователя, вне зависимости от особенностей его цветовосприятия [6].

Литература

6. Распространенность аномалий цветового зрения [Electronic resource] / Интернет-ресурс. - Режим доступа: www/ URL: http://www.everyday.com.ua/health/vision_anomaly.htm - Загл. с экрана.

7. Петровский, Б. В. Популярная медицинская энциклопедия, ст. “Зрение”, “Цветовое зрение”, ”Советская энциклопедия”, 1988

8. Особенности цветового восприятия человека [Electronic resource] / Интернет-ресурс. Режим доступа: www/ URL: <http://www.galactic.org.ua/Prostranstv/zrenie4.htm> - Загл. с экрана.

9. Восприятие мира и дальтонизм [Electronic resource] / Интернет-ресурс. - Режим доступа: www/ URL: <http://log-in.ru/articles/700/> - Загл. с экрана.

10. Цвет. Все о цвете [Electronic resource] / Интернет-ресурс. - Режим доступа: www/ URL: <http://color.ru/> - Загл. с экрана.

11. Цифровые технологии глазами дальтоника [Electronic resource] / Интернет-ресурс. - Режим доступа: www/ URL: <http://www.medkrug.ru/article/show/3807> - Загл. с экрана.