



ПРОСПЕКТ СВОБОДНЫЙ-2016

МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ СТУДЕНТОВ,
АСПИРАНТОВ И МОЛОДЫХ УЧЁНЫХ

ЭЛЕКТРОННЫЙ СБОРНИК МАТЕРИАЛОВ
МЕЖДУНАРОДНОЙ КОНФЕРЕНЦИИ СТУДЕНТОВ,
АСПИРАНТОВ И МОЛОДЫХ УЧЁНЫХ
«ПРОСПЕКТ СВОБОДНЫЙ-2016»,
ПОСВЯЩЁННОЙ ГОДУ ОБРАЗОВАНИЯ
В СОДРУЖЕСТВЕ НЕЗАВИСИМЫХ ГОСУДАРСТВ

КРАСНОЯРСК, СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

15-25 АПРЕЛЯ 2016 Г.

Министерство образования и науки Российской Федерации
ФГАОУ ВПО «Сибирский федеральный университет»

Сборник материалов
Международной конференции студентов,
аспирантов и молодых учёных
«Перспектив Свободный-2016»,
посвящённой Году образования
в Содружестве Независимых Государств

Красноярск, Сибирский федеральный университет, 15-25 апреля
2016 г.

Красноярск, 2016



ПЕРСПЕКТИВ СВОБОДНЫЙ-2016

МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ СТУДЕНТОВ, АСПИРАНТОВ И МОЛОДЫХ УЧЁНЫХ

Красноярск, Сибирский федеральный университет, 15-25 апреля 2016 г.

«Системный анализ, управление и программная инженерия»



**THE SYSTEM OF RATIONAL OPERATING UNITS MIX AT
HYDROPOWER PLANT. DEVELOPMENT OF CRITERIA ACCORDING TO
PERFORMANCE SPECIFICATION FOLLOWED BY INTRODUCTION AT UPPER
LEVEL OF AUTOMATED SYSTEM CONTROL OVER TECHNOLOGICAL
PROCESSES AT HYDROPOWER PLANT**

Rechkov A. V.

**Scientific supervisor the Head of technological control system service branch of
RusHydro PLC - Sayano-Shushenskaya HPP Sivtsov A. N.**

Language supervisor Tankov E. V.

Siberian Federal University

In the control process at an hydropower plant (HPP) a key factor is the automated processes control system (APCS), processing a huge amount of data from the various sensors, subsystems in real-time mode, and manages operating mode of hydroelectric power plant. In addition, APCS performs diagnostic functions, self-diagnostics, monitoring, regulating, archiving and storage of data, etc.

APCS consists of two levels: the upper-general and lower-aggregate one. The aggregate level controls basically the individual components of a hydraulic unit and general station level collects, processes and stores the information at the aggregate level, as well as information of the modular units, general station devices, switchgear. It executes a group of active power control functions (GAPC) and group regulation of voltage and reactive power (GRVRP).

Currently, at the most stations the composition of hydraulic units is defined by a station shift supervisor (SSS), because the APCS can not implement fully automatic control of all regime's objectives at the station. When deciding the SSS is guided by own knowledge and professional experience, but the human factor takes also the place, that is expressed in the subjectivity of perception, mental and emotional state, time pressure, because during decision making it is necessary to handle large amounts of incoming information in short space of time.

To increase the level of decisions taken by the SSS, it is necessary to automate the selection of hydraulic units composition. But HPP is a semistructured system that contains deterministic, stochastic and indefinite information about objects and processes. The process of selecting hydraulic units composition belongs to a group of multi-criteria and semistructured problems. Attempts to automate the process in this area began in the 70s of the 20th century. There have been developed such systems as the Autooperator, and ROUM later.

Autooperator was a worthy means of implementing HPP control mode for its time. But the experience of operating Autooperator in various hydroelectric plants has revealed significant shortcomings in their work.

To eliminate the shortcomings a new approach was to be applied. Conducted in the field research led to creation of the system known as rational operating units mix (ROUM). ROUM is a software-technical complex (STC) designed for determining the number of hydraulic units and their load in accordance with the given values of planned and unplanned load, the reserves of active and reactive power with minimum expenditure of energy carrier for 1 kW of generated power according to the expenditure of equipment service life.

Under composition is meant defining station numbers of units operating in generating mode and values of their loads.

STC is the basis for building automated systems at any power plant. STC should be focused on implementation of informational, control and auxiliary functions. STC should

provide reliability that is relevant to technical specification requirements for a particular process control system.

ROUM processes individual energy characteristics of the hydraulic units, the current state of the hydraulic units, especially electric circuits. It responds to scheduled and unscheduled power at an HPP. It means ROUM takes into account both operational and technical, as well as systemetic limitations.

Using the optimal composition of the hydraulic units can increase the efficiency of the plant by 3-5%, which will lead to an increase in power generation without additional use of water resources.

ROUM system has not been introduced because of imperfection of the STC in the 20th century. They did not allow handling a large volume of data in real time using sophisticated algorithms and schemes, so the introduction of the system has stopped.

Use of ROUM in an automatic control loop system is not very realistic at the moment, because the developed algorithms will not be able to cover fully the entire set of possible situations that arise in the course of control mode at an HPP, but ROUM can be an automated system. Applying operating in automatic mode ROUM system the operating personnel makes decisions to select a working mode of the HPP, but it receives data from ROUM - several options for the composition in order of reducing their profitability and indicating efficiency values. That allows the personnel to see the best, in controlling, composition of hydroelectric units. Thus ROUM when it's integrated into APCS upper-level of a power plant, which contains all the relevant information about the HPP can be used as an assistant to choose the optimal composition of units, but an SSS makes a decision: to use the information of ROUM system or not. However, even in that situation the system ROUM will be useful at an HPP to ensure its best operation.

References

1. Bryzgalov, V.I. *Gidroelektrostancii*: V.I. Bryzgalov, L.A. Gordon. - Krasnoyarsk: IPC KGTU, 2002, 541 s.
2. Mitrofanov, S.V. *Razrabotka sistemy podderzhki prinyatiya reshenij na osnove mnogokriterial'noj optimizacii sostava agregatov GEHS*: dis. kand. tekhn. nauk: 05.14.02: zashchishchena 24.04.14: utv. 13.03.2014/ Mitrofanov Sergej Vladimirovich. – N., 2013. – 213 s. –Bibliogr.: s.165–172.
3. RD 153-34.2-35.520-99 «Obshchie tekhnicheskie trebovaniya k programmno-tekhnicheskim kompleksam dlya ASU TP gidroelektrostancij».
4. Filippova, T.A. *Optimizaciya ehnergeticheskikh rezhimov gidroagregatov gidroelektrostancij*. : – M.: «EHnergiya», 1975. – 206 s.
5. Filippova, T.A. i dr.: *Gidroehtnergetika: uchebnik* / T.A. Filippova, M.SH. Misrihanov, YU.M. Sidorkin, A.G. Rusina. – Novosibirsk: Izd-vo NGTU, 2011. – 640 s. (Seriya «Uchebniki NGTU»).
6. Fomin, S.S. *Racional'noe upravlenie sostavom agregatov kaskada Bratskoj i Ust'-Ilimskoj GEHS* / S.S. Fomin, A.S. Misyul' : Racional'noe upravlenie predpriyatiem. #1/2014. str. 84-87.



АНАЛИЗ ВОЗМОЖНОСТЕЙ МОДЕЛИРОВАНИЯ ОРГАНИЗАЦИОННЫХ ОБЪЕКТОВ

Алфимов Д.Е.

научный руководитель д-р техн. наук, проф. Бронов С. А.

Сибирский федеральный университет

В современном мире все чаще встает задача повышения эффективности менеджмента в различных организациях. Для эффективного управления необходимо иметь модель управляемого объекта. В случае, если мы рассматриваем технические объекты, то мы можем их формализовать, составить математическое описание, а затем и математическую модель. Для большинства типов моделей технических объектов существует специальная теория. После составления модели нам остается корректно поставить цель, найти целевую функцию и решить задачу оптимизации при заданных ограничениях.

Однако, построение моделей организационных объектов на порядок сложнее из-за того, что они не так формализованы, а в большинстве операций полагаются на людей. Очень близко к формализации и построению моделей организационных объектов подходят промышленные ERP системы, ведущие учет ресурсов предприятия и поддерживающие принятие решений руководителями. Но в большинстве случаев, такие системы слишком специфичны, рассчитаны на одну предметную область или же рассматривает объект только с определенной точки зрения.

Если рассмотреть университет, с целью построения его модели, то мы можем выделить следующие его функции: обучающая, исследовательская, профессиональная, сервисная, культурная и гуманистическая [3]. Помимо этих функций в университете выполняются работы по обеспечению его деятельности всеми необходимыми ресурсами. За каждую функцию в структуре университета отвечает некоторое подразделение, которое в свою очередь может делиться на меньшие подструктуры.

Таким образом, мы можем представить университет в виде иерархической системы, каждый элемент которой осуществляет определенные задачи и имеет свою зону ответственности.

Это очень удобный способ управления сложностью, когда каждому структурному уровню соответствует уровень решаемых задач. На более высоких уровнях мы можем не учитывать некоторые детали работы более низких уровней. Так, например, нет необходимости поднимать вопрос составления учебного плана конкретной специальности на уровень дирекции университета.

Таким образом университет можно представить системой, составленной из подсистем, эмерджентные свойства которых обеспечивают выполнение основных задач, стоящих перед университетом. Особо хочется подчеркнуть тот факт, что эмерджентные свойства систем обусловлены их структурой, так как из одних и тех же составных блоков мы можем получить системы с различными свойствами [1].

Однако, прежде чем изучать варианты построения структур, необходимо рассмотреть на основании каких данных принимаются решения. Вышестоящие подразделения получают информацию от нижестоящих, тогда необходимо сначала смоделировать нижестоящие уровни.

Начать моделирование можно с некоторого подразделения. Представим её в виде иерархической системы, с одним руководителем и несколькими подчиненными - исполнителями. Подчиненные выполняют задачи, согласно своим профессиональным обязанностям, в порядке некоторого приоритета, используя доступные ресурсы. Стоит

отметить, что руководитель не является единственным источником задач, так как в ходе работы могут появляться дополнительные задачи.

Руководитель несет ответственность за работу подразделения, он планирует деятельность, учитывая необходимые для этого ресурсы, стараясь равномерно распределить нагрузку между подчиненными. Здесь стоит отметить, что в качестве критерия эффективности могут быть использованы разные характеристики: равномерная загрузка персонала, оборудования, максимальная производительность в условиях ограниченных ресурсов. Соответственно для достижения действительно оптимального управления нам необходимо построить модель такой иерархической системы с несколькими исполнителями.

Как было сказано выше, подчиненные выполняют свои обязанности, но каждый из них может справляться по-разному, даже если они получают одно и то же задание, так как они подвержены влиянию таких факторов как мотивация, профессионализм, навыки, умения, психологические особенности конкретного человека. Так же достаточно сложно оценить затраты в случае некоторой творческой работы, когда трудно оценить количество необходимых этапов, не говоря уже об оценке необходимых ресурсов. Тогда руководителю при планировании придется учитывать не только объемы предстоящей работы с учетом выделенных ресурсов, но и эти факторы тоже. В этом заключается основная сложность и основная причина неточности в управлении подразделением. Руководитель может опросить подчиненных, о том сколько займет та или иная работа, но люди не всегда точно могут оценить предстоящий объем работы. Но именно опираясь на эти данные о затратах времени и других ресурсов, руководитель должен скоординировать работу подразделения, чтобы исключить лишние простои.

Очень широко эта тема обсуждается в сообществе разработчиков программного обеспечения. Разработка программного обеспечения — это весьма творческий процесс, а учитывая то, что требования к готовому продукту могут изменяться во время работы над ним, а заказчиком выступает бизнес, для которого проект, законченный позже срока равносильен провалу, то проблема оценки затрат ресурсов здесь стоит еще острее.

Специально для этого разработаны специальные методологии разработки программного обеспечения, такие как scrum и канбан. Они помогают формализовать процесс, стандартизировать задачи и построить модель работы команды с определенной точностью. Основными ресурсами в этой отрасли является время, трудозатраты на разработку. Тогда опираясь на данные информационной системы, руководитель команды может узнать приблизительное количество задач необходимых для завершения проекта, среднее количество задач, возникающих в процессе работы и может построить функцию распределения вероятности завершения проекта в определенное время. Затем выбрать степень лояльности к риску и, по графику, определить время завершения проекта.

Тот же подход можно применить и для других ресурсов, в иных отраслях. Если руководитель будет использовать информационную систему для организации распределения задач, подчиненные будут своевременно вносить изменения о ходе выполнения задач, то на основании этих данных можно будет смоделировать рассмотренное подразделение.

Список литературы

1. Тарасенко Ф.П. Прикладной системный анализ. М.: Кнорус, 2010. 224 с.
2. Структура СФУ [электронный ресурс]. Режим доступа: <http://structure.sfu-kras.ru/>
3. Огородникова И.А. Идея университета - проект воплощения идеальной образовательной формы // Вестник Омского университета. 2014. №4. С. 77-80.



ИСПОЛЬЗОВАНИЕ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ РАСПОЗНАВАНИЯ РЕЧИ ПО ГУБАМ

Антипов В. Г.

научный руководитель канд. техн. наук Царев Р. Ю.

Сибирский федеральный университет

В настоящее время задачи распознавания образов являются очень распространёнными. Одной из таких задач стало чтение по губам, которая возникает в различных сферах жизни. Система, умеющая распознавать речь по видеозаписи движения губ, была бы очень полезна в криминалистике. В условиях, когда аудиодорожка очень зашумлена, повреждена или отсутствует совсем, но необходимо получить информацию из записанных разговоров, единственным выходом является чтение по губам. Также, такая система была бы полезна для людей с ограниченными возможностями.

Постановка задачи для распознавания речи по губам: имеется видеозапись с речью человека, необходимо распознать её и представить в текстовом виде. При решении такой задачи в первую очередь необходимо понимать, саму технику чтения по губам и, непосредственно, составляющие речи – звуки (фонемы) и движения губ, соответствующих этим звукам [1].

Для системы, способной читать по губам, можно выделить следующие функциональные блоки:

- выделение контура губ;
- выделение определённых движений губ;
- соотнесение движений губ со звуками.

В основе системы распознавания речи по губам, а именно для соотнесения движений губ со звуками, может лежать искусственная нейронная сеть. Искусственная нейронная сеть (ИНС) – математическая модель, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма [2]. ИНС состоит из элементов, называемых искусственными нейронами, который имеют ряд функциональных возможностей схожих с элементарными функциями биологического нейрона. Искусственные нейроны организуются аналогично (но не всегда) тому, как это происходит с нейронами в человеческом мозгу.

Несмотря на довольно поверхностное сходство, ИНС демонстрирует удивительные свойства присущие мозгу [3]. Одно из таких свойств – способность нейросети обучаться, т.е. под воздействием внешней среды менять своё поведение. Это свойство является ключевым для ИНС, ведь от принципов, методов, корректности и скорости обучения зависит адекватность функционирования модели. Если в системе реализован метод обучения, то она сможет обучаться на одних примерах, а затем распознавать другие, отличные от примеров обучающей выборки. Кроме того, подавая системе в качестве обучающей выборки примеры речи на различных языках, можно перенастроить систему для любого языка.

Существующие на данный момент системы чтения по губам имеют достаточно невысокий процент успешного распознавания. Связано это с рядом причин, в первую очередь с тем, что некоторые звуки похожи по движениям губ, тогда как другие вовсе не имеют видимой артикуляции.

Таким образом, исследуемая задача является одной из самых сложных для искусственного интеллекта. Использование нейросетевой модели для соотнесения

движения губ и звуков, благодаря способности к обучению, делает систему более адаптированной к новым данным.

Список литературы

1. Компьютер научили читать по губам [Электронный ресурс] // Интернет-издание «N + 1». – Режим доступа: <https://nplus1.ru/news/2016/03/28/leap-reading-machine>.
2. Искусственная нейронная сеть [Электронный ресурс] // «Википедия» – свободная энциклопедия. – Режим доступа: <http://ru.wikipedia.org>.
3. Уоссермен, Ф. Нейрокомпьютерная техника: Теория и практика / Ф. Уоссермен; перевод на русский язык Ю.А. Зуев, В.А. Точенов. – Москва: Мир, 1992. – 102 с.



**ИДЕНТИФИКАЦИЯ ПРОЦЕССА РАСПРЕДЕЛЕНИЯ ДАВЛЕНИЯ
ЛИНЕЙНОЙ ЧАСТИ МАГИСТРАЛЬНОГО НЕФТЕПРОВОДА****Антропов Н.Р.****научный руководитель канд. техн. наук Агафонов Е.Д.***Сибирский государственный аэрокосмический университет им. акад.**М.Ф. Решетнева*

Основной характеристикой любого измерительного прибора (датчика) является погрешность измерения контролируемого параметра. Погрешность измерения отображает величину расхождения между результатом измерения и реальным значением измеряемого параметра, и указывается в техническом паспорте прибора. Однако, на практике, величина погрешности часто не соответствует заявленным данным, а иногда и вовсе неизвестна. При этом в процессе эксплуатации прибора величина погрешности постоянно меняется. Причинами изменения погрешности могут стать изменение условий эксплуатации (внешние факторы), дрейф физических параметров самого датчика (внутренние факторы), возмущения в каналах связи датчика с регистрирующей аппаратурой, частичный или полный выход из строя датчика и так далее. В связи с этим, разработка соответствующих алгоритмов коррекции погрешностей является актуальной задачей, причем очевидна необходимость построения адаптивных подходов, функционирующих в условиях неопределенности и изменчивости окружающей среды.

Объектом исследования в настоящей работе является магистральный нефтепровод. Современные трубопроводные сети и гидравлические системы оснащены различной измерительной аппаратурой и информационно-измерительными системами, использующимися для мониторинга технологических параметров. Одним из наиболее важных технологических параметров трубопровода является давление, и от того, насколько точно оно известно, зависит эффективность и надежность эксплуатации технологических систем и оборудования трубопровода.

В научной и справочной литературе [1-3] предлагаются различные подходы к коррекции погрешностей данных измерений датчиков. Первый подход состоит в конструктивном изменении и оптимизации устройства датчика (оптимизация свойств чувствительно элемента, топологии, конструкции корпуса). Второй подход заключается во включении в структуру прибора дополнительных пассивных и активных элементов, обеспечивающих необходимую коррекцию. И третий подход заключается в разработке алгоритмических методов коррекции по математической модели (концепция интеллектуального датчика). В рамках настоящей работы рассматривается третий подход.

Сформулируем постановку задачи. Вдоль профиля линейной части магистрального нефтепровода на неравномерном расстоянии друг от друга располагаются датчики давления. Датчики давления состоят из чувствительных элементов, воспринимающих давление, и преобразователей, собранных в корпусе. По причинам, связанным с различием в типах, производителях датчиков, их технического состояния, особенностей считывания данных измерений, измеренные величины имеют как различные систематические погрешности, так и случайные погрешности с различными статистическими характеристиками (законами распределения, параметрами распределения и др.). Необходимо, пользуясь сведениями о физических закономерностях распределения давления вдоль профиля трубопровода, сведений о взаимном расположении датчиков, а также набора измеренных величин (выборки)

скорректировать показания датчиков, т.е. уменьшить величины их погрешностей (случайная аддитивная, мультипликативная помехи и систематическая погрешность).

В соответствии с уравнением Бернулли [4], давление на участке трубопроводе при установившихся режимах его работы падает линейно. Линейность распределения давлений это идеальное приближение для случая постоянства внутреннего сечения трубопровода, неизменности физических характеристик перекачиваемой нефти вдоль профиля трубопровода и т. д. В реальности распределение давлений несколько отличается от линейного. Наличие местных сопротивлений, естественный дрейф физических характеристик нефти способствуют сдвигу линии гидравлического уклона. Поэтому, измерения датчиков давлений, расположенных вдоль профиля нефтепровода также необходимо использовать для полноты модели.

Существующие на данный момент алгоритмические подходы зачастую ограничены использованием в них классических методов анализа данных в рамках параметрического подхода. А именно, данные обрабатываются с применением моделей с известной параметрической структурой и последующей оценкой параметров, например в [3]. Такие процедуры не позволяют с достаточной степенью полноты использовать всю имеющуюся априорную информацию. В связи с этим для решения поставленной задачи предлагается использовать гибридные модели вида [5-6], как средства наиболее полного учета априорной информации о процессе измерения и характеристик измеряемых величин.

Построение гибридных моделей производилось по следующему алгоритму. Пусть имеется выборка наблюдений состояния объекта $D_s: V=\{u(t), x(t), t=\overline{1, N}\}$ и известны сведения $D_p: \hat{A}(x, \alpha)$ о виде преобразования $A(x)$ с точностью до набора параметров α (линейная модель).

Примем $q_\Sigma(t)=A(u(t))-\hat{A}(u(t))$ либо $q_\Pi(t)=A(u(t))/\hat{A}(u(t))$, тогда комбинированная модель запишется в виде:

$$x_\Sigma(t) = \hat{A}(u(t), \alpha) + \hat{q}(t),$$

или

$$x_\Sigma(t) = \hat{A}(u(t), \alpha) \hat{q}(t).$$

Построение параметрической модели $x(t)=\hat{A}(u(t))$ может производиться с использованием достаточно хорошо разработанных методов параметрической идентификации на основании априорных сведений о структуре оператора $\hat{A}(u(t))$. В данной работе использовался метод наименьших квадратов.

Преобразование $\hat{q}(t)$ восстанавливается с помощью непараметрической оценки регрессии [8]:

$$\hat{q}(t) = \frac{\sum_{t=1}^N q[t] \prod_{k=1}^m \Phi\left(\frac{u_k - u_k[t]}{c_n}\right)}{\sum_{t=1}^N \prod_{k=1}^m \Phi\left(\frac{u_k - u_k[t]}{c_n}\right)},$$

где значения $q[t]$, $t=\overline{1, N}$ вычисляются на основании выборки $V=\{u(t), x(t), t=\overline{1, N}\}$ по следующим формулам:

$$q_\Sigma[t] = x(t) - \hat{A}(u(t), \alpha), t = \overline{1, N}$$

$$q_\Pi[t] = \frac{x(t)}{\hat{A}(u(t), \alpha)}, t = \overline{1, N}$$

Ниже, в таблице 1, представлены результаты расчета среднеквадратической ошибки идентификации для различных моделей.

Таблица 1. Результаты расчета среднеквадратической ошибки

Тип модели	Среднеквадратическая ошибка
Непараметрическая регрессия	6.823e-04
Параметрическая линейная модель	19.8128
Гибридная модель с аддитивной невязкой	1.486e-04
Гибридная модель мультипликативной невязкой	7.859e-04

Наименьшая величина ошибки у гибридной модели с аддитивной невязкой, что объясняется учетом дополнительных сведений и обеспечением полноты используемой априорной информации. На следующем рисунке представлено сравнение аппроксимационных свойств непараметрической регрессии и гибридной модели.

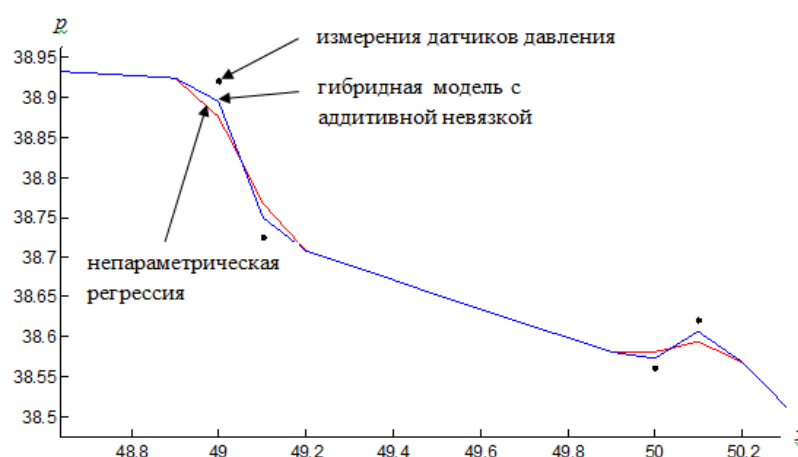


Рис.1 - Сравнение аппроксимационных свойств непараметрической регрессии гибридной модели с аддитивной невязкой

Таким образом, применение предложенных моделей позволит повысить точность моделирования соответствующих трубопроводных систем и, как следствие, качество их управления.

Список литературы

1. Бромберг Э.М. Тестовые методы повышения точности измерений / Э. М. Бромберг, К. Л. Куликовский. – М. : Энергия, 1978. – 176 с.
2. Гельман М.М. Автоматическая коррекция систематических погрешностей в преобразователях «напряжение – код» / М.М. Гельман, Г.Г. Шаповал. – М. : Энергия, 1974. – 88 с.
3. Земельман М.А. Автоматическая коррекция погрешностей измерительных устройств / М. А. Земельман. – М. : Изд-во стандартов, 1972. – 199 с.
4. Лурье М.В. Математическое моделирование процессов трубопроводного транспорта нефти, нефтепродуктов и газа / М.В. Лурье – М.: Издательство «Нефть и газ» РГУ нефти и газа им. И.М. Губкина, 2003. – 335 с.
5. Медведев А. В. Теория непараметрических систем. К-модели // Вестник СибГАУ. Вып. 3(36). Красноярск. 2011. С. 57-62.
6. Лапко В.А. Синтез и анализ гибридных моделей стохастических зависимостей в условиях наличия их частного описания // Автометрия. 2004. Т. 40, № 1. С. 51-59.

ПРОГРАММНАЯ СИСТЕМА ПОСТРОЕНИЯ МАРШРУТА ТОРГОВОГО ПРЕДСТАВИТЕЛЯ

Бридский А.Е.

научный руководитель канд. техн. наук Даничев А.А.

Сибирский федеральный университет

Задача магистерской работы состоит в том, чтобы составить оптимальный маршрут для торгового представителя.

Для расчета оптимального маршрута торгового представителя, используются самые актуальные данные (маршрут, время и информация о пробках онлайн), полученные с помощью сервиса Yandex.API, далее применяется алгоритм расстановки торговых точек по предпочтительному времени посещения торговой точки, на основе задачи коммивояжера. В результате работы программы получили статистику по времени работы в каждой торговой точке, а также по времени, которое было затрачено в пути до торговой точки. На основе полученных статистических данных рассчитаны персональные коэффициенты по времени в пути и работы в каждой торговой точке, для каждого сотрудника персонально применен свой коэффициент.

Объект изучения – исследование вариантов оптимизации маршрутов торгового представителя.

Целью является разработка, обоснование и реализация программно-алгоритмического комплекса, обеспечивающего формирование ежедневного формирования маршрутного листа, предназначенного для работы торгового представителя.

Знаменитая задача коммивояжера, поставленная еще в 1934 г., является одной из самых важнейших задач в теории графов. В своей области (оптимизации дискретных задач) задача коммивояжера служит своеобразным полигоном, на котором испытываются все новые методы.

Постановка задачи. Коммивояжер (бродячий торговец) должен выйти из первого города, посетить по разу в неизвестном порядке города $2, 3, 4, \dots, n$ и вернуться в первый город. Расстояния между всеми городами известны. В каком порядке следует обходить города, чтобы замкнутый путь коммивояжера был кратчайшим? В терминах теории графов: найти гамильтонов цикл в графе минимальной длины.

Задача коммивояжера является так называемой NP-трудной задачей, т.е. задачей, точное решение которой в общем случае может быть получено только за экспоненциальное время. Следовательно, решать ее переборным алгоритмом неэффективно при большом количестве вершин графа.

Проведено исследование программных продуктов, которые позволяют решать проблему построения маршрутов. В ходе анализа, выяснилось, что разработанные программные продукты разные по своему назначению и функционалу, требуют тонкой настройки, либо имеют некорректный алгоритм работы, из-за этих факторов возникает множество проблем с построением маршрута.

В ходе решения задачи о составлении оптимального маршрута для торгового представителя, был разработан следующий алгоритм:

Вход: сотрудник, точки, которые нужно посетить, и время для любых двух точек, требуемое для переезда из одной точки в другую; информация о точках, содержащаяся в массиве points, такая как время, с которого можно посещать точку (visitFrom), время, до которого следует посетить точку (visitTo). Значение const, которое является результатом интенсивных исследований, оценивающее

приблизительное время, в среднем затрачиваемое перевозчиком на доставку товара из одной точки в другую.

Выход: конкретные моменты времени, в которые необходимо посетить все точки, уложившись в заданные ограничения, либо сообщение о том, что для данного набора точек невозможно построить расписание.

Алгоритм:

1. Упорядочиваем точки в порядке возрастания времени, с которого можно посещать каждую точку, а если оно совпадает у каких-либо двух точек, то в порядке возрастания времени, до которого их можно посещать.

2. Вычисляем время для переезда из точки в точку с учетом коэффициента сотрудника, если результирующее время попадает в интервал времени работы этой точки, то заносим время в массив. Если полученное время находится слева от интервала времени, в который нужно посетить данную точку, то прибавляем к времени величину, на которую требуется перенести точку, чтобы она попадала в этот интервал и заносим в массив. В противном случае ищем в массиве точек такую, после которой можно вставить точку, чтобы она попадала в свой промежуток времени и чтобы величина, на которую сдвигались точки, была минимальной (minDiff). Если находим такую точку, то вставляем точку $i + 1$ после этой точки, и сдвигаем все элементы в массиве timeline на величину minDiff , попутно проверяя, попадают ли точки в свои промежутки времени. Если какая-либо точка не попадает, то завершаем алгоритм с отрицательным результатом. Выполняем эту процедуру для всех точек.

3. Далее запускается процедура оптимизации. Для всех точек ищем группы, время работы которых пересекается и решаем для этих групп задачу коммивояжера. Формируем новое решение на основании полученных решений для пересекающихся групп. Завершаем алгоритм.

Список литературы

1. Карасев, А.И. Курс высшей математики для экономических вузов: Учебник для экономических вузов // А.И. Карасев, З.М. Аксютин, Т.И. Савельева - М.: Высшая школа, 1982. - С.279-285.

2. Полунин, И.Ф. Курс математического программирования: Учебник для вузов// - Минск: Высшая школа, 1970. - С. 194-230.

3. Лопатин А. П. Моделирование перевозочного процесса на городском пассажирском транспорте // А. П. Лопатин. – М.: Транспорт, 1985. – 144 с.

4. Котиков Ю. Г. Основы системного анализа транспортных систем: учебное пособие // Ю. Г. Котиков. – СПб.: СПбГАСУ, 2001. – 264 с.

5. Писецкая В.В. Логистика автомобильного транспорта: магистерская работа // Режим доступа: <http://masters.donntu.edu.ua/2002/fem/pisetskaya/diss.htm>

6. Ганущак Н.К. Исследование существующих алгоритмов решения транспортных задач в ГИС: магистерская работа // Режим доступа: <http://www.masters.donntu.edu.ua/2006/ggeo/ganushchak/diss/index.htm>

7. Семенов В.В. Математическое моделирование транспортных потоков мегаполиса // Режим доступа: <http://spkurdyumov.narod.ru/Semenov.pdf>



СИНТАКСИЧЕСКАЯ МОДЕЛЬ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ С ОБЩИМ СЕМАНТИЧЕСКИМ БАЗИСОМ

Валуцкая Э.А.

научный руководитель канд. техн. наук Кузнецов А.С.

Сибирский федеральный университет

Как известно, на практике иногда применяются, или встречаются программные средства, основанные на многоязычности в смысле общей семантической базы. То есть структура современных языков имеет общую семантику (начальное смысловое значение операторов, основных конструкций языка и т.д.), но при этом имеет разный синтаксис (набор правил, описывающий комбинации символов алфавита) и разную лексику (правила образования цепочек символов). Иначе говоря, современные языки программирования дают разработчику идентичные возможности, при разном внешнем виде программ.

При сравнении подобных структур языков программирования с точки зрения семантики можно сделать акцент на основные компоненты конструкций, которые фигурируют в часто используемых современных языках программирования. Эти компоненты и есть основа распространенных языков программирования, которая прошла проверку временем и практикой.

Описание общей семантической базы современных языков программирования может дать уникальную возможность для генерации компилятора, который будет работать с любым из существующих языков программирования, у которых общая семантическая база. Таким образом, единая семантика позволит пользоваться общим семантическим анализатором и общим генератором исполняемого кода для всех используемых языков.

Современные языки программирования похожи друг на друга, идентичность их далеко не полная. Каждый содержит структуры, присущие только ему. Однако существует общая семантическая зона, в которую входят конструкции, принадлежащие всем языкам программирования или большей части языков. Таким образом, семантику каждого языка программирования можно условно поделить на «область пересечения» (структуры общие для всех языков программирования) и «область объединения» (структуры присущие только одному языку программирования). Поэтому создание входного языка для многоязычного компилятора можно произвести двумя различными способами[1]:

1. Использовать только общие конструкции (область пересечения), отбросив все «особенные» конструкции языков, как не необходимые. Это приведет к усечению всех языков программирования.

2. Использовать все имеющиеся в языках конструкции (область объединения). В этом случае каждый из языков должен быть дополнен конструкциями, имеющимися в других языках программирования. Этот подход чреват чрезмерным расширением семантической базы.

Наличие общей семантической базы сделает перевод с одного языка программирования на другой тривиальным (простейшим), так как изменения будут касаться только внешнего вида, а содержание будет неизменно. На базе универсального компилятора может быть создана многоязыковая интегрированная среда разработки программ, в которой в качестве внутреннего представления программ будут использоваться не тексты, а семантические структуры. В такой среде понятие языка

разработки превратится в условность, поскольку переключение с одного языка на другой можно осуществить в любой момент выбором соответствующего пункта меню.

Также для создания многоязыкового компилятора достаточно написать несколько front end'ов. Front end – это начальная стадия, или как часто ее называют анализ. Анализ разбивает исходную программу на составные части и накладывает на них грамматическую структуру. Затем он использует эту структуру для создания промежуточного представления исходной программы. Если анализ обнаруживает, что исходная программа неверно составлена синтаксически, либо дефектна семантически, он должен выдать информативные сообщения об этом, чтобы пользователь мог исправить обнаруженные ошибки. Анализ также собирает информацию об исходной программе и сохраняет ее в структуре данных, именуемой таблицей символов, которая передается вместе с промежуточным представлением синтезу[2].

Другими словами это та часть, которая занимается разбором входного текста на языке программирования и строит в некотором виде образ исходной программы - так называемое, промежуточное представление.

Рассмотренная выше методология может быть воплощена на практике, такие трансляторы существуют [1]. Однако они имеют следующие связанные друг с другом недостатки:

1. Статичность архитектуры транслятора. При необходимости добавления нового языка или обновления существующего (например, после принятия очередного стандарта) выполнить это можно только путем полной пересборки проекта. Кроме того при реализации и тестировании понадобится выполнить много «ручной работы».

2. Представленная методология предполагает использование универсальных языков программирования, семантика которых ограничена средой времени выполнения. В то же самое время, на практике часто используются предметно-ориентированные языки. Кроме того, нередко недостаточными оказываются не все, а подмножество конструкций универсального языка, называемых лингвистами подъязыками. В обоих случаях семантика ограничена не средой выполнения, а языком программирования.

Обойти первое ограничение можно с использованием класса мультисинтаксических языков программирования [3].

Есть язык-лидер – язык высокого уровня, есть конструкции, включающие «фрагменты» кода на языке (включаемый) в код на лидере. Такой языково-ориентированный мультисинтаксический подход позволяет в пределах одного исходного текста программы одновременно использовать языки с различающимися свойствами с точки зрения синтаксиса и, как следствие, генерируемые разными классами грамматик.

При этом синтаксис каждого из составляющих языков описывается контекстно-свободной грамматикой. В грамматике языка-лидера выделяются специальные символы, сигнализирующие «главному» синтаксическому анализатору о передаче управления соответствующему «вспомогательному» анализатору. Количество лексических и семантических анализаторов может быть равно количеству используемых языков, а их взаимодействие друг с другом осуществляется по одной из традиционных схем (ассемблерные вставки, скриптовые языки, встраивание языков запросов к базам данных в языки программирования). Взаимодействие анализаторов представлено на рисунке 1.

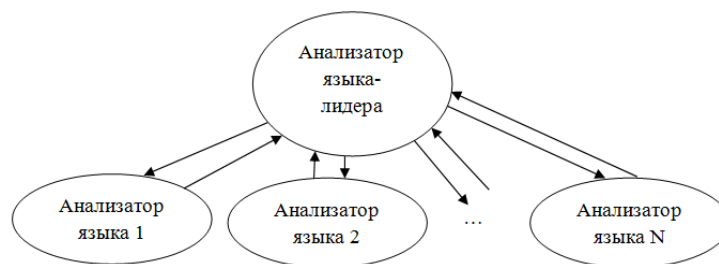


Рис.1 – Взаимодействие анализаторов

В качестве одного из наиболее известных способов оптимизации по быстродействию программ, написанных на языке высокого уровня, можно привести прямое включение ассемблерного кода в текст программы. Использование ассемблера, встроенного в языки высокого уровня, для создания мультиверсионного ПО позволяет обеспечить мультиверсионность на уровне групп инструкций. Иначе говоря, разработчик может реализовать одну версию фрагмента кода на языке высокого уровня, например, C/C++, а вторую функционально-эквивалентную – в виде ассемблерной вставки.

При выполнении определенных условий такой подход позволяет добиться мультиверсионности на уровне функций. Для этого тело какой-либо версии метода класса или функции полностью реализуется в виде ассемблерной вставки (исключения могут составить объявления объектов, локальных для функции или метода), а другая на лидирующем языке. Такой подход, очевидно, применим не только для мультиверсионного ПО, но и для других классов.

К настоящему моменту не существует трансляторов языков программирования, допускающих включения ассемблерного кода более чем одной системы команд, то есть двух и более платформ. Располагая такой возможностью, можно было бы, во-первых, конструировать версии одного и того же модуля для разных платформ, в том числе для устаревших или вновь разрабатываемых, имея «под рукой» уже работающие версии. Во-вторых, можно было бы конструировать программные модули для одной платформы, но с использованием ассемблеров разных стилей [4].

Современные компиляторные технологии позволяют реализовывать синтаксические анализаторы включаемых языков в виде проходов (как в LLVM) или подключаемых модулей (как в GCC), поэтому разработка единого компилятора языков с общим семантическим базисом не должна составить большой проблемы. В настоящее время ведутся дальнейшие исследования в этом направлении.

Список литературы

1. Эволюция современных языков программирования [электронный ресурс]. Режим доступа: <http://www.osp.ru/pcworld/2001/03/161246/>;
2. Ахо, Альфред В., Лам, Моника С, Сети, Рави, Ульман, Джеффри Д., Компиляторы: принципы, технологии и инструментарий, 2-е изд. : Пер. с англ. - М. : 000 "И.Д. Вильямс", 2008. - 1184 с. : ил. - Парал. тит. англ.;
3. Кузнецов, А.С. Генерация компиляторов мультисинтаксических языков программирования мультиверсионных систем / А.С. Кузнецов, И.В. Ковалев // Международный журнал «Программные продукты и системы». – 2008. – Вып. 4 (84). – С. 101-103.
4. Кузнецов, А.С. Разработка мультиверсионных программ с использованием мультисинтаксических языков и технологий [Текст] / А.С. Кузнецов // Вестник НИИ СУВПТ: Сб. науч. трудов. Красноярск: НИИ СУВПТ, 2008. – № 26. – С. 25-42.

ПОЛЕ ЗНАНИЙ ДЛЯ ВЫБОРА МЕТОДА УПРАВЛЕНИЯ КОСМИЧЕСКИМ АППАРАТОМ

Зюзина В. И., Космынина Н. А.

научный руководитель д-р техн. наук. Князькин Ю. М.

Сибирский федеральный университет

Современный космический аппарат (КА) – дорогостоящая, сложная, многофункциональная система, удалённая от Земли на значительное расстояние и функционирующая в условиях агрессивной внешней среды.

Управление космическим аппаратом – процесс поддержания его в работоспособном состоянии, а также обеспечения выполнения целевой задачи, с момента запуска на орбиту и на протяжении всего срока эксплуатации [1].

Для управления автоматическими КА применяются следующие методы управления:

1. Командный метод управления – метод, при котором каждая конкретная операция управления осуществляется только по командам от наземного комплекса управления [2, 8].

2. Программный метод управления – метод, который реализуется на борту космического аппарата с использованием программ, закладываемых с Земли, и выполняемых далее автономно [2, 8].

3. Комбинированное управление или командно-программный метод управления – метод управления, при котором производится контроль отработки бортовой программы и оперативное вмешательство в процесс управления при отклонении его от требуемого [2, 8].

4. Программно-временной метод управления – метод управления, который обеспечивает отработку программ, во время которых подаётся управляющее воздействие в определённый момент времени [3, 8].

5. Автономный метод управления осуществляется с помощью бортовой цифровой вычислительной машины, входящей в состав бортовой аппаратуры и выдающей определённую команду на основе анализа текущих условий, в соответствии с заложенной ранее логикой [4, 8].

6. Координатный метод управления реализуется с помощью управляющих воздействий, закладываемых на борт КА и выдаваемых по достижению нужных координат [5, 8].

7. Координатно-временной метод управления характеризуется тем, что управляющие команды (сигналы) формируются как функция не только измеряемых, но и прогнозируемых параметров движения центра масс КА [5, 8].

8. Координатно-программный метод управления заключается в реализации программ, заложенных с Земли на борт КА и выдаваемых по достижению нужных координат [5, 8].

Вместе с совершенствованием космической техники всё более усложняется процесс управления, поэтому важно на этапе проектирования выбрать соответствующий метод управления (или их совокупность). В настоящее время метод управления выбирается на основе опыта и знаний экспертов предметной области. Представление такого типа знаний осуществляется с помощью разработки экспертных систем. Перед разработкой экспертной системы необходимо построить «поле знаний», что и является целью данной статьи.

Поле знаний – это неформальное описание основных понятий и взаимодействий между понятиями предметной области, выявленных из системы знаний экспертов, в виде графа, диаграммы, таблицы или текста. В данной статье поле знаний будет построено в виде графа. Граф будет состоять из трёх уровней:

1. Первый уровень – входные данные будущей экспертной системы.

2. Второй уровень – промежуточные параметры экспертной системы, полученные при сочетании входных данных.

3. Третий уровень – выходной параметр экспертной системы.

Для определения значений выходного параметра необходимо, чтобы уровень обработки входных значений корректно отображал картину логических процессов, проходящих на этапе проектирования.

На первом уровне будет пять вершин: «Назначение КА», «Срок активного существования», «Этап жизненного цикла КА», «Тип орбиты» и «Периодичность нахождения КА в зоне радиовидимости» (рисунок 1).

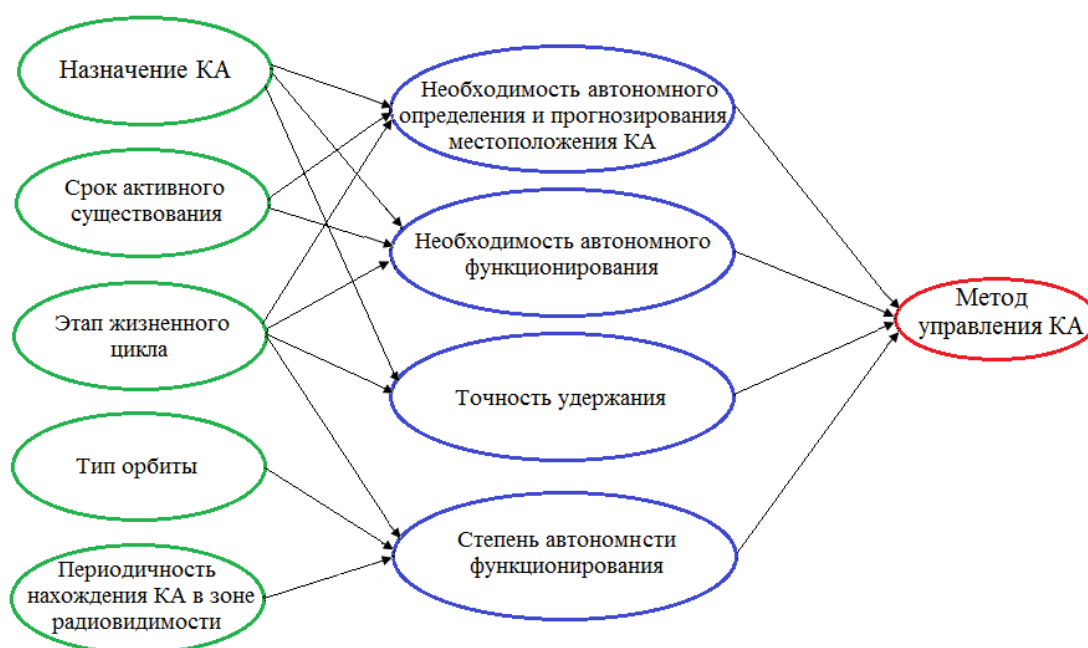


Рис.1 – Поле знаний для выбора метода управления КА

Первой вершиной является «Назначение КА», которое включает в себя все целевые задачи, решаемые проектируемым КА. Данная вершина может принимать следующие значения [6]:

1. «КА связи, телевещания и ретрансляции», предназначенные для передачи различного типа информации на большие расстояния между абонентами.

2. «КА навигации и времени», предназначенные для оперативного навигационно-временного обеспечения неограниченного числа пользователей.

3. «Геодезические КА», предназначенные для создания единой системы координат всей поверхности земного шара, установления геодезических связей между континентами и островами, уточнения геофизических параметров Земли.

4. «КА земного мониторинга», предназначенные для наблюдения за состоянием земель и природных ресурсов, своевременного выявления различных изменений, их оценки, а также предупреждения и устранения последствий негативных процессов.

5. «КА изучения дальнего космоса», предназначены для исследования межпланетного пространства и небесных тел.

Вторая вершина – «Срок активного существования», который показывает, длительность исполнения КА целевых задач. Может принимать следующие значения:

1. «До 1года».
2. «От 1 года до 5лет».
3. «Более 5лет».

Следующая вершина – «Этап жизненного цикла КА». Он введён для уточнения метода управления и принимает значения:

1. «Выведение» КА на орбиту, близкую к расчётной.
2. «Приведение КА в штатную орбитальную позицию» – устранение ошибок выведения и стабилизации полёта КА в рабочей орбитальной позиции, оценка внешних возмущающих моментов, влияющих на параметры орбиты.
3. «Орбитальный манёвр».
4. «Поддержание полёта КА в рабочей орбитальной позиции» при эксплуатации аппарата по целевому назначению.
5. «Увод КА с заданной орбитальной позиции на орбиту «захоронения»» и полёт спутника по ней.

Вершина «Тип орбиты» может принимать следующие значения [6]:

1. «Низкая круговая орбита» – орбита до 2000км.
2. «Средняя круговая орбита» – орбита до 20000км.
3. «Высокоэллиптическая орбита» – орбита, у которой высота в апогее во много раз превышает высоту в перигее.
4. «Геосинхронная орбита» – орбита, на которой период обращения КА равен периоду обращения Земли.

Значения вершины «Периодичность нахождения КА в зоне радиовидимости» будут следующие:

1. «Более 5 витков в сутки».
2. «Менее 5 витков в сутки».
3. «Менее 1 витка в сутки».

На втором уровне сочетания вершин первого уровня формируют четыре вершины: «Необходимость автономного определения и прогнозирования местоположения КА», «Необходимость автономного вычисления», «Точность удержания КА на орбите» и «Степень автономности функционирования» (рисунок 1).

Вершина «Необходимость автономного определения и прогнозирования местоположения КА» означает наличие системы, позволяющей измерять и предсказывать параметры движения центра масс КА в любой момент времени. Данная вершина имеет следующие значения [7]:

1. «Есть такая необходимость».
2. «Такой необходимости нет».

Следующая вершина – «Необходимость автономного вычисления» принимает следующие значения:

1. «Есть такая необходимость».
2. «Такой необходимости нет».

Вершина «Точность удержания КА на орбите» является следующим параметром в экспертной системе для определения метода управления и может принимать значения [7]:

1. «Точность удержания высока».
2. «Низкая точность удержания».

Вершина «Степень автономности функционирования» указывает, сколько по времени КА находится в зоне радиовидимости. В зависимости от этого может принимать значения [6]:

1. «Непрерывная» – КА постоянно находится в зоне радиовидимости.
2. «Сеансная» – КА находится в зоне радиовидимости периодически в определённые моменты времени.

На третьем уровне будет только одна вершина – «Метод управления», который имеет следующие значения:

1. «Командный метод управления».
2. «Программный метод управления».
3. «Командно-программный метод управления».
4. «Командно-временной метод управления».
5. «Автономный метод управления».
6. «Координатный метод управления».
7. «Координатно-временной метод управления».
8. «Координатно-программный метод управления».

Построенное поле знаний для выбора метода управления космическим аппаратом на этапе проектирования изображено на рисунке 1. Из рисунка видно, при сочетании каких вершин первого уровня получается каждая из вершин второго уровня. А совокупность всех вершин второго уровня определяет значение единственной вершины третьего уровня.

Построенное поле знаний может использоваться для разработки экспертной системы в сфере выбора метода управления автоматическими КА на этапе проектирования.

Список литературы

1. Кравец В. Г. Автоматизированные системы управления космическими полётами – М.: Машиностроение, 1995 – 254с.
2. Лапушкин В. Н. Основы управления космическими аппаратами в полёте: Акционерное общество «Информационные Спутниковые Системы» имени академика М. Ф. Решетнёва, 2012. – 382с.
3. Использование сеансно-временного метода при обращении к спутнику в задаче мониторинга животных в заповедниках. [Электронный ресурс]. – Режим доступа: <http://knowledge.allbest.ru/ecology>, свободный (дата обращения 18.10.2015).
4. Управление полётом космического аппарата. [Электронный ресурс]. – Режим доступа: <http://www.cosmos-journal.ru/articles/936/>, свободный (дата обращения 18.10.2015).
5. Научно-технические исследования и практические разработки ГНП РКЦ «ЦСКБ-Прогресс» Космическое аппаратостроение. – Самара: 2011 – 280с.
6. Чеботарёв В. Е. Основы проектирования космических аппаратов информационного обеспечения: учеб. пособие / В. Е. Чеботарёв, В. Е. Косенко; Сиб. гос. аэрокосмич. ун-т. – Красноярск, 2011. – 488с.
7. Чеботарёв В. Е. Проектирование космических аппаратов систем информационного обеспечения. Книга 2. Внутреннее проектирование космического аппарата – Красноярск.: СибГАУ им. академика М. Ф. Решетнёва, 2005 – 168с.
8. Зюзина В. И. Методы управления автоматическими космическими аппаратами/ Зюзина В. И., Космынина Н. А.// Молодежь и современные информационные технологии (Том 1)– 2016 – С. 253-254.



СЕГМЕНТАЦИЯ ИЗОБРАЖЕНИЯ НА ОСНОВЕ АНАЛИЗА ОБЛАСТЕЙ КАК ОДИН ИЗ СПОСОБОВ ОПРЕДЕЛЕНИЯ ГРАНИЦ МЕЖДУ ЛЮМЕНОМ И СТЕНКОЙ КЛЕТКИ ХВОЙНЫХ ПОРОД ДЕРЕВЬЕВ

Кипель Н.А.

научный руководитель кан. техн. наук, доц. Дьячук П.П.

Сибирский федеральный университет

На рост деревьев влияют многие внутренние и внешние факторы, воздействие которых находит свое отражение в клеточной структуре. Это делает деревья уникальным объектом исследования и предоставляет ученым возможность на основе анализа клеточной структуры буквально образом заглянуть в прошлое. Исследования, основанные на «памяти деревьев» позволяют установить связь между условиями окружающей среды и древесной структурой в определенный момент времени. Такими исследованиями занимается наука дендрохронология. Дендрохронология – это систематическое изучение древесных колец с целью датирования событий прошлого и оценки климатических изменений [1]. Основателем и организатором систематических дендрохронологических исследований являлся американский астроном А.Дуглас в начале 20 века. Он собирал и анализировал образцы «желтой сосны» для изучения колебаний ее годовичных приростов и установления связи с циклами солнечной активности.

Структуру деревьев можно изучать как в целом, так и на клеточном уровне. Самым распространенным параметром является количество годовых колец, которое говорит о возрасте дерева. Еще один немаловажный параметр – ширина годичного кольца, которая говорит о благоприятности условий его формирования. Однако, изучив клеточную структуру можно получить гораздо больше информации об условиях окружающей среды в определенный момент времени. На клеточном уровне важны такие параметры, как длина и ширина клетки, ее площадь, размеры люмена и стенки клетки. На рисунке 1 представлена клетка хвойной породы дерева с указанием всех ее параметров.

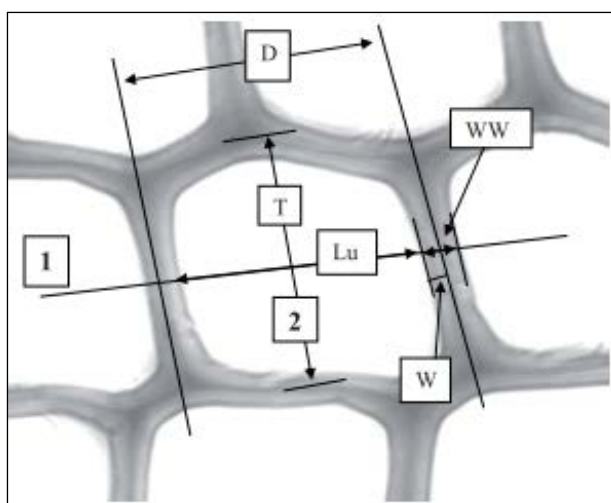


Рис.1 – Клеточная структура хвойных пород деревьев

- 1) Радиальный размер люмена – Lu;
- 2) Радиальный размер клетки – D;

- 3) Тангенсальный размер клетки – T ;
- 4) Толщина клеточной стенки – W ;
- 5) Толщина двойной клеточной стенки – WW [2].

Одной из основных проблем в дендрохронологии состоит в нехватке оборудования и программного обеспечения, способного проводить исследования с высокой точностью. При проведении исследований человеческий фактор является решающим, именно от него зависит качество полученных образцов, снимков клеточной структуры, вычислений макро- и микропараметров. При развитии современных технологий необходимо уделить внимание автоматизации данных процессов. В настоящее время в лабораториях дендрохронологии используют программное обеспечение, которое поставляется с микроскопами и является многофункциональным и позволяют решить общие задачи по вычислению микропараметров. На данный момент существует несколько адаптированных модулей для работы с клеточной структурой дерева. Эти модули построены на базе программного обеспечения AxioVision и ImageProPlus и они являются применимыми в дендрохронологии, однако, требуют некоторой доработки для увеличения точности вычислений. Получение качественного снимка образцов клеток зависит от работы оператора и его настройки микроскопа. Очевидно, что для обеспечения высокой точности и исключения человеческого фактора более целесообразно применять автоматические настройки в зависимости от исходных образцов. Качественно сделанные снимки – это залог дальнейшей успешной работы. При настройке важно подобрать оптимальные цветосветовые характеристики, чтобы не потерять информацию. После получения снимков в цифровом виде они требуют дополнительной предварительной обработки, которая необходима для того, чтобы сократить потерю информации и скорректировать цветосветовые параметры изображения.

После предварительной обработки начинается основной этап вычисления параметров клеток. Этот этап также во многом зависит от качества образцов и снимков. Если образцы исследования – срезы клеточной структуры дерева или же их снимок на микроскопе были сделаны не качественно, то производить вычисления будет намного сложнее и невозможно будет получить точные результаты без погрешностей. Главная проблема при вычислении параметров клеток дерева – это точное определение границы стенки и люмена клетки. Суть проблемы состоит в том, что при переходе из одного цвета в другой, в данном случае это переход от серого к черному, от серого к белому и наоборот возникают граничные пиксели, которые имеют схожие значения яркости, которые трудно обнаружить человеческому глазу. При установлении этой границы оператором вручную дальнейшие вычисления не будут обладать необходимой точностью. Исходя из этого, целью работы является поиск оптимального метода нахождения границ, задача сводится к сегментации изображения.

Существует несколько основных методов сегментации изображения: методы на основе анализа пикселей, областей, контуров и на основе моделей и т.д., каждый из этих методов направлен на решение определенных задач распознавания. Методы на основе анализа пикселей не рассматривают окрестность пикселя, а оперируют только значения яркости отдельных пикселей. Методы на основе анализа областей анализируют уровни яркости пикселей в более крупных областях. Методы на основе анализа контура выделяют контур объекта и стараются ему следовать. Применение методов на основе моделей возможно лишь в тех случаях, когда нам известна геометрическая форма объекта. Клеточные структуры не имеют определенной геометрической формы, поэтому данный метод не возможно применить. В нашем случае считается более целесообразным применять методы на основе анализа областей, так как необходимо принимать во внимание связность пикселей. Рассмотрим

подробнее метод связывания пирамид, так как он является эффективным методом комбинирования алгоритма сегментации и вычисления признаков.

На первом этапе строится пирамида Гаусса. При составлении пирамиды Гаусса строка яркости пикселей сканируется окном, длина которого составляет 4 пикселя, что обусловлено четырехсвязным соседством пикселей. Каждый последующий отсчет проходит с шагом 2, что говорит о возможности отнесения пикселя к двум областям. На каждом шаге вычисляется среднее арифметическое для вычисления элементов верхней строки. Каждая строка содержит вдвое меньше элементов, чем предыдущая. Данная процедура продолжается до тех пор, пока пирамида не будет завершена одним элементом. Полученная пирамида называется пирамидой Гаусса. Каждый элемент строки называется узлом пирамиды, которые соединены ребрами.

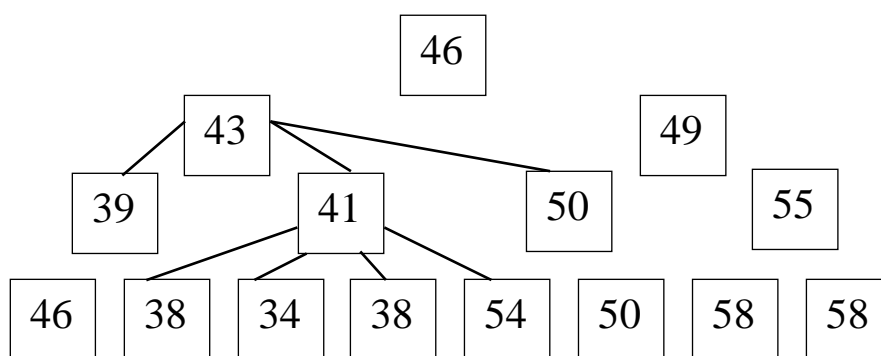


Рис.2 – Построение пирамиды Гаусса

Следующий этап – это связывание пирамиды. Суть данного этапа заключается в перестроении связей элементов. Для пикселей нижнего уровня находят пиксели второго уровня, которые имеют более близкие значения. Каждый пиксель может иметь только одну связь с вышележащим пикселем. В результате получается новая структура данных, именуемая деревом.

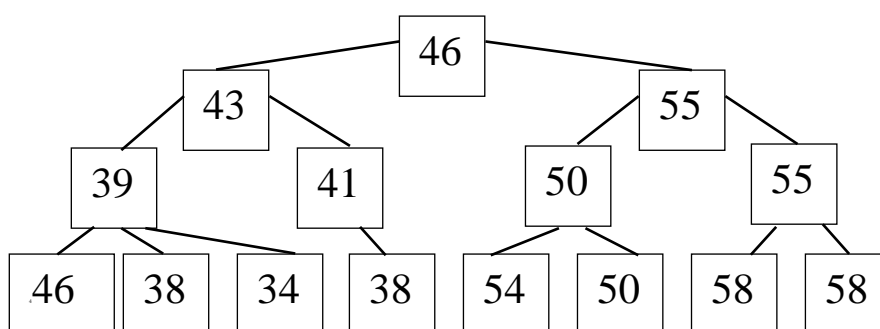


Рис.3 – Связывание узлов

Усреднение связанных узлов является третьим шагом алгоритма. На этом шаге алгоритма полученная структура связей используется для перерасчета средних уровней яркости на этот раз с учетом новых связей. Среднее значение на вышележащем уровне рассчитывается исходя из значений нижних связанных пикселей. Данная процедура повторяется до вершины дерева. Последние два шага повторяются друг за другом до тех пор пока не будет достигнут конечный устойчивый результат. Окончательный результат заключается в «спуске» значений двух вершин поддеревьев к самому

нижнему уровню, проходя по всем вершины пирамиды, данные значения указываются в скобках, таким образом отчетливо видно разделение на две области.

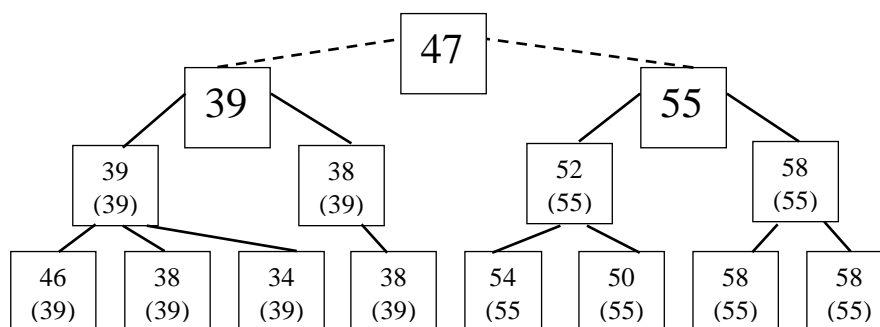
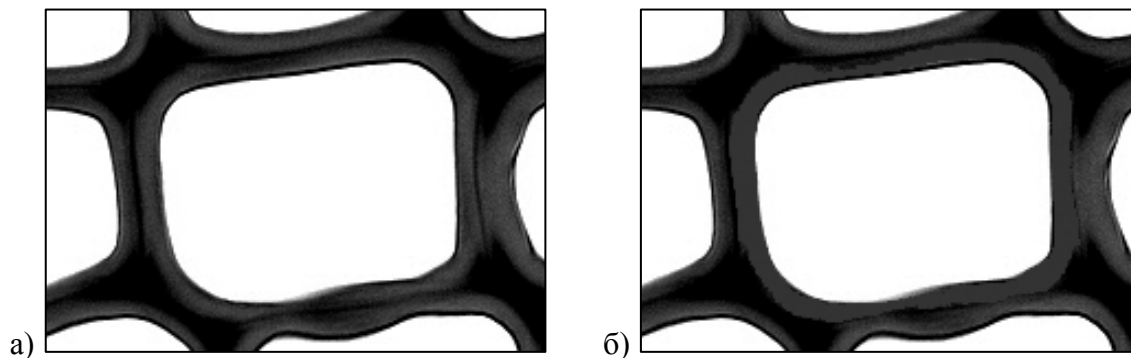


Рис.4 – Окончательный результат работы алгоритма

Алгоритм сегментации на основе областей был применен для изображений клеточной структуры хвойных пород деревьев. Результат работы алгоритма представлен на рисунке 5.



**Рис.5 – Применение алгоритма к изображениям клеточной структуры:
а) зашумленная оболочка клетки б) результат сегментации**

Список литературы

1. Ваганов Е.А. Методы дендрохронологии: учебное пособие/ Ваганов Е.А., Круглов В.Б., Васильев В.Г. – Дендрохронология – Красноярск, 2008. – 120 с.
2. Силкин П.П. Методы параметрического анализа структуры годичных колец хвойных: монография, 2010. – 335 с.
3. Яне Б. Цифровая обработка изображений – Мир цифровой обработки - Техносфера, 2007. – 584с.

МЕТОДИКА ОЦЕНКИ ТРУДОЕМКОСТИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Коротких И.Ю.

научный руководитель канд. техн. наук Якунин Ю.Ю.

Сибирский федеральный университет

Некачественное планирование часто является причиной неудачи многих проектов. Особенно это справедливо для проектов разработки программного обеспечения. Временные затраты на разработку программного обеспечения практически напрямую зависят от его размера и трудоемкости разработки. Применение существующих методов оценки разработки ПО на практике оказывается весьма трудоемким, кроме того, они недостаточно формализованы для учёта разнообразных факторов, влияющих на трудоемкость и длительность проекта.

В самом общем смысле, все методы оценки проектов разделяются на две группы: микрооценки и макрооценки.

Главной проблемой современных методов оценки трудозатрат является сложность их адаптации к конкретному проекту. Обобщённые модели, разработанные с учетом данных о большом количестве проектов, не позволяют в достаточной для планирования степени учесть этой специфики. В существующих методах оценки трудоемкости уделяется мало внимания или не уделяется совсем влиянию психологических аспектов, как индивидуальных участников, так и всей команды. Тем не менее, эта составляющая оказывает существенное влияние на точность оценки.

Как правило, разработка программного обеспечения – сложный и долгий процесс, в котором задействовано большое количество людей. Выделим основные технические роли и должности, которые существуют сегодня в разработке ПО: программист, ведущий программист, системный архитектор, системный аналитик, тестировщик, системный администратор, руководитель проекта

Специфика психологических свойств личности, влияющих на успешность трудовой деятельности, очевидна. В современной психологии нет единого понимания личности, но существует множество способов ее исследования. Общеизвестный параметр – типология.

Типология – система анализа личности, разработанная швейцарским психиатром К. Г. Юнгом в его работе «Психологические типы» [1], опубликованной в 1921 году. К.Г. Юнг в своих работах, изучая и анализируя личность и психическую деятельность человека, выделил 8 психологических типов людей. Изначально Юнг разделил людей на две большие группы по доминирующей ориентации: интровертной или экстравертной.

Кроме того, Юнг выделяет основные психологические функции: мышление (Т), чувство (Ф), ощущение (S) и интуиция (N). Все 4 присутствуют в каждом человеке, но одна из них является доминирующей. Мышление и чувство Юнг отнес к разряду рациональных (J) психологических функций. Ощущение и интуиция, по его мнению, относятся к разряду иррациональных (P) функций

Таким образом, четыре функции можно разбить попарно на два подкласса: как человек воспринимает окружающий мир и чем человек пользуется, совершая свои поступки в нем. То есть человек либо сенсорик, либо интуит и, либо логик, либо этик. Тогда мы можем перейти к плодам работы ученых, которые в дальнейшем развивали топологические мысли К. Г. Юнга. Они разделили людей уже на 16 типов, представив каждый тип четырехбуквенной формулой: первая пара букв E/I говорит об

эгоориентации (экстраверт или интроверт), вторая S/N - о восприятии человеком окружающего мира (сенсорик или интуит), третья T/F — чем человек руководствуется в своих действиях (логик или этик), и наконец четвертая пара J/P — о том, какая функция преобладает (рациональная или иррациональная).

В статье [2] предлагаются четыре цветовых идентификатора психологических типов людей, которые распределены по принципу психологических установок экстраверсии и интроверсии (E/I) и классов, рациональных и иррациональных (J/P). Все типы разбиваются на восемь диад, состоящих из четырех основных цветовых секторов, каждый из которых (для простоты понимания) обозначен своим цветом (на рис. 1).

Экстраверты иррациональные сведены в зеленый сектор. Экстраверты рациональные – в красный. Интроверты рациональные – синий сектор и интроверты иррациональные – желтый сектор. В каждом из четырех секторов с большой вероятностью можно утверждать, что в нем находится вектор профессионального развития человека в соответствии с его генетическими предрасположенностями, задатками, наклонностями и способностями. В результате тестирования автором статьи [2] было выявлено, что наиболее эффективная деятельность человека находится в своем цветовом секторе.

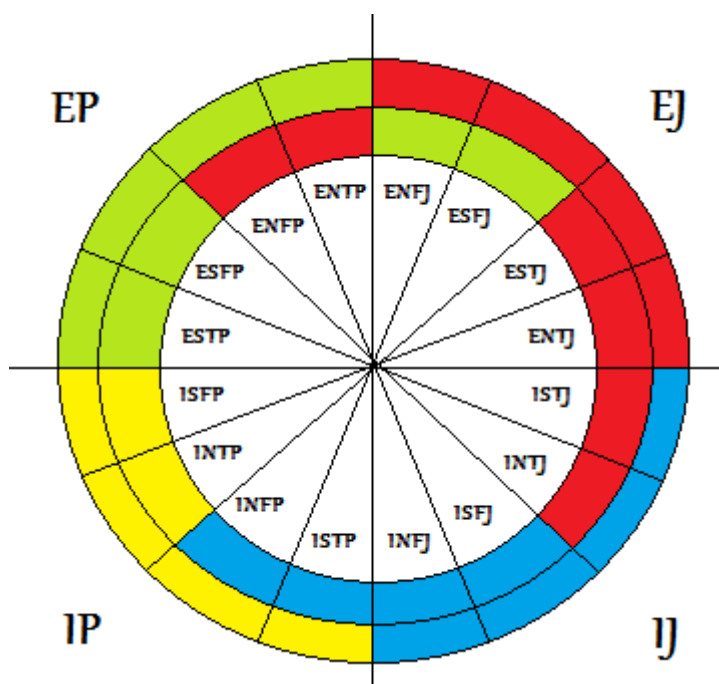


Рис.1 – Сектора, соответствующие психологическим типам

Психотип красного сектора обладает лидерскими качествами и способен к стабильному развитию и управлению системой. Люди из зеленого сектора, обладая инновационно-креативным мышлением, способны к организации коммуникативных связей и продвижению идей в массы. Люди из синего сектора предрасположены к размеренной усидчивой деятельности, способны руководить стабильной устойчивой системой. Люди из желтого сектора обладают способностью к глубокому обдумыванию результатов, анализу выполненной работы, глубокому творческому погружению. Руководствуясь принципами, изложенными в [3], поставим в соответствие определенному виду деятельности в команде разработчиков ПО психотип как показано в таблице 1.

Таблица 1- Пример соответствия видов деятельности типам личности

Вид деятельности	Основная задача	Психологический тип
Программист	Непосредственно отвечает за то, что программа работает правильно (в соответствии с техническим заданием).	ISFJ
Ведущий программист	Занимается составлением технических заданий для программистов и нередко их обучает. Также выполняет особо сложную и важную работу по кодированию.	ISTJ
Системный архитектор	Активно занимается исследованиями и экспериментами, и документирует архитектурные решения.	INTP
Системный аналитик	Переводит пожелания заказчика в формат точно описанных технических заданий.	INTJ
Тестировщик	Всеми возможными и невозможными способами пытается сломать программу и найти в ней ошибки и проблемы.	ENTJ
Системный администратор	Отвечает за работу сети, серверов, компьютеров разработчиков и программ.	ESTP
Руководитель проекта	Соединяет работу технической части команды разработки с другими отделами фирмы и “высоким” руководством.	ESTJ

В зависимости от подхода в организации к описанию и формализации требований к программной системе и на основе анализа технического задания можно выделить функциональные и нефункциональные требования. От функциональных требований трудоемкость разработки системы зависит в большей степени, хотя некоторые нефункциональные требования могут очень сильно влиять на стоимость разработки. В общем случае для любых подходов функциональные требования к системе на самом высоком уровне могут быть описаны как требования к подсистемам. Каждая подсистема может состоять из набора функций (по ГОСТ 34.602–89), которые имеют детальное описание или спецификацию. Очевидно, что нельзя не учитывать методологию, инструменты и среду разработки ПО, которые могут в большей степени повлиять на длительность и трудоемкость программного проекта. В рамках данного исследования за основу взяты гибкие (agile) методологии [4], основные идеи которых состоят в следующем:

1. Люди и их взаимодействие важнее процессов и инструментов;
2. Готовый продукт важнее документации по нему;
3. Сотрудничество с заказчиком важнее жестких контрактных ограничений;
4. Реакция на изменения важнее следования плану.

Предлагаемый метод расчета трудоемкости основан на макрооценке функциональных точек [5-6] и учитывает функциональные требования, компетенции и психотипы разработчиков (рис. 1).



Рис.1 – Исходные данные для расчета трудоемкости

Здесь F – множество функциональных точек, $R_1 \dots R_m$ – разработчики, $Skill(R_i)$ – навыки разработчиков, $Ps(R_i)$ – психотипы разработчиков, T – итоговая трудоемкость программного проекта.

Укрупненный алгоритм оценки трудоемкости состоит из следующих шагов.

- 1) Определение множества функциональных точек на базе анализа требований и анализа объекта автоматизации.
- 2) Тестирование кандидатов в команду проекта и определение их компетенций и психотипов.
- 3) Выбор методологии и среды разработки. Выбор инструментов.
- 4) Распределение исполнителей по ролям.
- 5) Распределение задач по исполнителям.
- 6) Расчет трудоемкости на базе математической модели и статистических данных о предыдущих проектах.

Для реализации данного алгоритма ведутся работы по созданию и исследованию математической модели, учитывающей психологические особенности членов команды разработчиков.

Список литературы

1. Юнг, К. Г. Психологические типы / под ред. В. Зеленского; пер. С. Лорие. — Азбука, 2001.
2. Краснов И.З. Дифференцированный подход к подготовке специалистов информационной безопасности / И.З Краснов // Решетневские чтения. — 2013. — Т. 2. — № 17. — С. 301-304.
3. Гуленко В. В. Менеджмент слаженной команды / В.В. Гуленко. – Астрель, 2003. – С. 282.
4. Кон М. Гибкая разработка ПО / М. Кон. — Вильямс, 2011. — С. 576.
5. Якунин, Ю.Ю. Оценка трудоемкости разработки программной системы / Ю.Ю.Якунин // Вестник Сибирского государственного аэрокосмического университета имени академика М.Ф. Решетнева. – Красноярск: СибГАУ, 2008. – Вып. 2(19). – С. 87-91с.
6. Варламов, А. С. Методика оценки трудоемкости выполнения программного проекта // Молодёжь и наука: Сборник материалов VIII Всероссийской научно-технической конференции студентов, аспирантов и молодых учёных, посвященной 155-летию со дня рождения К. Э. Циолковского [Электронный ресурс]. — Красноярск: Сибирский федеральный ун-т, 2012. — Режим доступа: <http://conf.sfu-kras.ru/sites/mn2012/section12.html>, свободный.

ОСОБЕННОСТЬ РЕАЛИЗАЦИИ СЕРВИС-ОРИЕНТИРОВАННОЙ АРХИТЕКТУРЫ ПРИ ИСПОЛЬЗОВАНИИ ASP.NET MVC FRAMEWORK

Костюк А.В., Васильев Э.В.

научный руководитель канд. пед. наук, доц. Виденин С.А.

Сибирский федеральный университет

В современном мире все очень динамично, и бизнес не исключение. С каждым годом бизнес-процессы уже функционирующего предприятия меняются все чаще и чаще, появляются новые услуги, компании диверсифицируются и выходят на новые рынки. Вслед за этими переменами изменениям подвергаются и корпоративные информационные системы (КИС). Из этого следует, что современная КИС должна быть гибкой и масштабируемой.

Перед нами была поставлена задача разработать ИС для центра дополнительного и безотрывного образования по ИТ ИКИТ. На старте мы выявили несколько потенциальных проблем при расширении приложения:

- бизнес-процессы не до конца поставлены;
- бизнес-процессы будут меняться в ближайшем будущем.

Следовательно, необходимо определиться с архитектурой, которая позволит нам безболезненно внедрять изменения в ИС.

Мы рассмотрели несколько подходов к разработке КИС: традиционный и сервис-ориентированный. Первый подход реализовывается следующими этапами:

- исследование предметной области;
- разработка архитектуры системы;
- реализация проекта;
- внедрение системы;
- сопровождение системы.

При таком подходе встречается дублирование кода, потому что логика пишется для каждого бизнес-процесса отдельно, и некоторые ее части могут повторяться. Следовательно, даже при малейшем изменении бизнес-логики приходится переписывать большую часть кода, что требует огромное количество времени и ресурсов.

Второй подход подразумевает, что наше приложение будет состоять из набора независимых сервисов, каждый из которых реализует отдельную бизнес-функцию, которая является логически обособленной, повторяющейся задачей, являющейся составной частью бизнес-процесса предприятия. Более того, сервисы могут быть реализованы независимо от языков программирования и других технических особенностей реализации, что дает возможность использовать различные технологии и фреймворки. Также сервисы могут быть написаны в независимости от других служб системы, необходимо только знание интерфейса используемых сервисов, то есть службы будут слабосвязны (loose coupling).

Мы прогнозируем изменения в КИС в ближайшее время. Например, учиться будут не только студенты, но и работники предприятий. В перспективе допустим выход системы как продукта (система для повышения квалификации). Поэтому выбор пал на сервис-ориентированную архитектуру.

Иными словами, мы выделили следующие достоинства данного подхода:

- отсутствие дублирования кода;
- независимость от языка программирования;
- низкая связанность.

В ходе реализации мы столкнулись с технической особенностью ASP.NET MVC, а именно: в данной технологии используются строго типизированные представления, т.е. такие представления, которые работают с определенной моделью. Однако, в большинстве случаев нам нужны не все поля из источника данных, и требуются другие, например, вычисляемые. Поэтому возникла необходимость использования собственной клиентской модели. Данные, которые приходят извне, конвертируются посредством сервисов в нашу модель. Это позволяет нам писать логику в контроллерах и верстать представления независимо от того, какие данные мы изначально получаем.

Учитывая все вышеперечисленные особенности, мы решили совместить подход EAI (Enterprise Application Integration) и SOA (Service-Oriented Architecture). Суть подхода EAI Hub/Spoke заключается в использовании централизованного брокера (Hub) и адаптеров (Spoke). Адаптеры соединяют сторонние приложения, которые используют свои источники данных, с хабом. Spoke присоединяются к приложениям и преобразовывают данные приложения в формат, который понимает хаб и наоборот. В нашем случае адаптерами являются сервисы, конвертирующие получаемые данные к формату нашей модели. В роли хаба выступает наше централизованное приложение. Это и позволяет решить поставленную задачу.

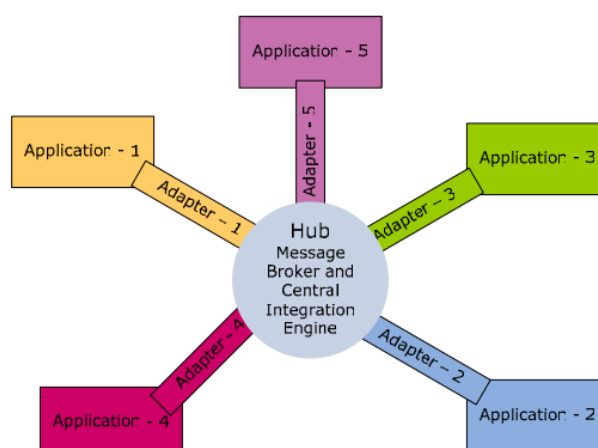


Рис.1 – EAI Hub/Spoke архитектура

В результате комбинации двух подходов EAI и SOA, мы получили следующие преимущества:

- требуется меньше ресурсов и снижаются временные затраты на внесение изменений в информационную систему;
- при появлении новых бизнес-процессов нам не придется дублировать код для их реализации. В некоторых случаях, для создания нового бизнес-процесса, достаточно скомбинировать уже существующие сервисы;
- независимость от источника данных, т.к. работа производится с собственной, или клиентской, моделью.

Список литературы

1. Anurag Goel. Enterprise Integration EAI vs. SOA vs. ESB.
2. Service Oriented Architecture (SOA) [электронный ресурс]. Режим доступа: <https://msdn.microsoft.com/en-us/library/bb833022.aspx>
3. Service Oriented Architecture: What Is SOA? [электронный ресурс]. Режим доступа: <http://www.opengroup.org/soa/source-book/soa/soa.htm>

РАЗРАБОТКА ИНТЕГРИРОВАННОГО РЕШЕНИЯ ИНФОРМАЦИОННОЙ ПОДДЕРЖКИ ПРОЦЕССОВ ПРОИЗВОДСТВЕННОГО ПЛАНИРОВАНИЯ

Котова М.В., Винниченко Д.И., Черномаз Р.И.
научный руководитель канд. техн. наук Капулин Д.В.
Сибирский федеральный университет

Эффективным методом повышения производительности труда является использование системы сетевого планирования для оптимизации стратегического плана развития предприятия, который служит основой оперативного управления комплексом производственных и технологических работ. Основным плановым документом в такой системе является сетевой график (сеть) представляющий собой информационно-динамическую модель, в которой отражаются все логические взаимосвязи и результаты выполняемых работ, необходимых для достижения конечной цели стратегического планирования. Область разработки и исследования процессов сетевого планирования не достаточно хорошо проработана, прикладные решения внедряются сложно, по причине того, что реализация плана производства на предприятии достигается совместной скоординированной деятельностью всех структурных подразделений. Для достижения ожидаемого результата работы предприятия нужно заранее выявлять факторы, отрицательно влияющие на нормальный ход производственного процесса. Это обстоятельство вызывает необходимость обеспечения постоянного текущего контроля над ходом производства, выявления причин, оказывающих отрицательное влияние на него, и принятия мер по регулированию технологического процесса ^[1].

Использование сетевых методов позволяет поднять качество и координацию действий, зависящих от многих факторов, связанных с получением и обработкой информации, рациональным распределением ресурсов, построением взаимоотношений смежных подразделений. Основным результатом сетевого планирования является календарный план, в котором комплексные задания разбиваются на отдельные, расположенные в технологической и временной последовательности, работы. Такой оптимизированный календарный план способствует минимизации длительности выполнения проектных и производственных работ за счет высокой координации деятельности различных подразделений предприятия ^[2].

Известные аналогичные решения по автоматизации процессов планирования не могут учесть большое количество одновременно обрабатываемых заказов с широкой номенклатурой изделий. С другой стороны, существующие теоретические методы позволяют представить производственный процесс в виде четкой согласованной последовательности, а объектно-ориентированный подход позволяет задать функционал для этой последовательности. В качестве программной платформы предлагаемого решения сетевого планирования (рис. 1) используется архитектура .NET Runtime Environment (DNX) и построенная на нем ASP.NET Core 1.0. Такое решение дает возможность копировать все необходимые библиотеки вместе с приложением в существующее окружение, а также запустить .NET Core кроссплатформенно на Linux и Mac OS. Приложения обращаются к серверу баз данных MS SqlServer 2008 R2, который используется исходя из требований технического задания, с использованием языка T-SQL для получения исходной информации о производстве. Web-приложения предприятия, реализующие бизнес-логику, разрабатываются под архитектуру ASP.NET Web Forms и ASP.NET MVC и размещаются на сервере приложений IIS. При этом

часть логики и расчетов, которые относятся к визуализации, реализуются на языке JavaScript с применением библиотеки JQuery.

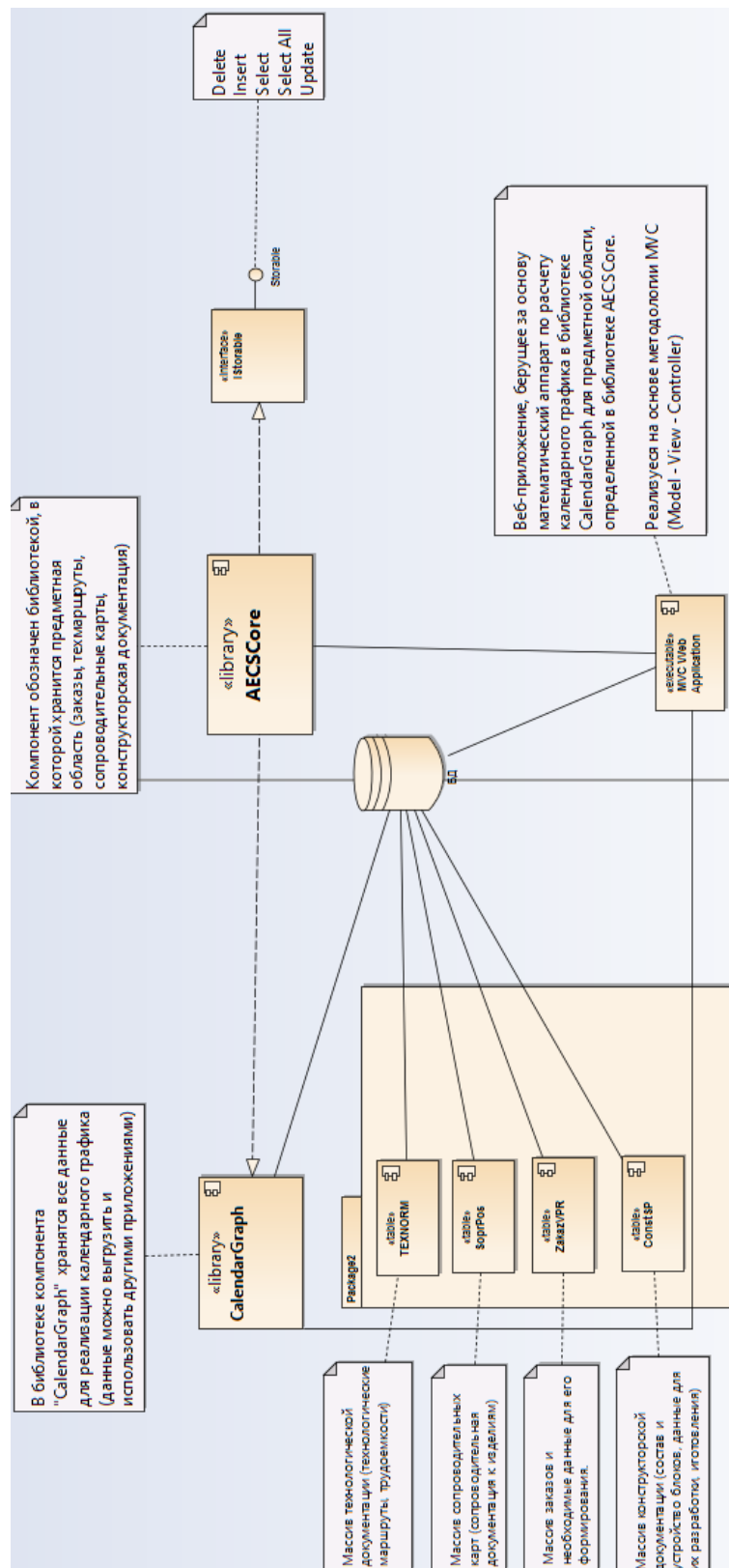


Рис.1 - Архитектура системы производственного сетевого планирования

Архитектура ASP.NET MVC часто используется для построения архитектурного каркаса, когда переходят от теории к реализации в конкретной предметной области. Тем самым разделяется ответственность за различное функционирование программных модулей. Для отображения сетевой модели предполагается использовать ленточную диаграмму (диаграмму Ганта), которая будет строиться с использованием библиотеки DHTMLX Basic Gantt Chart (рис. 2).

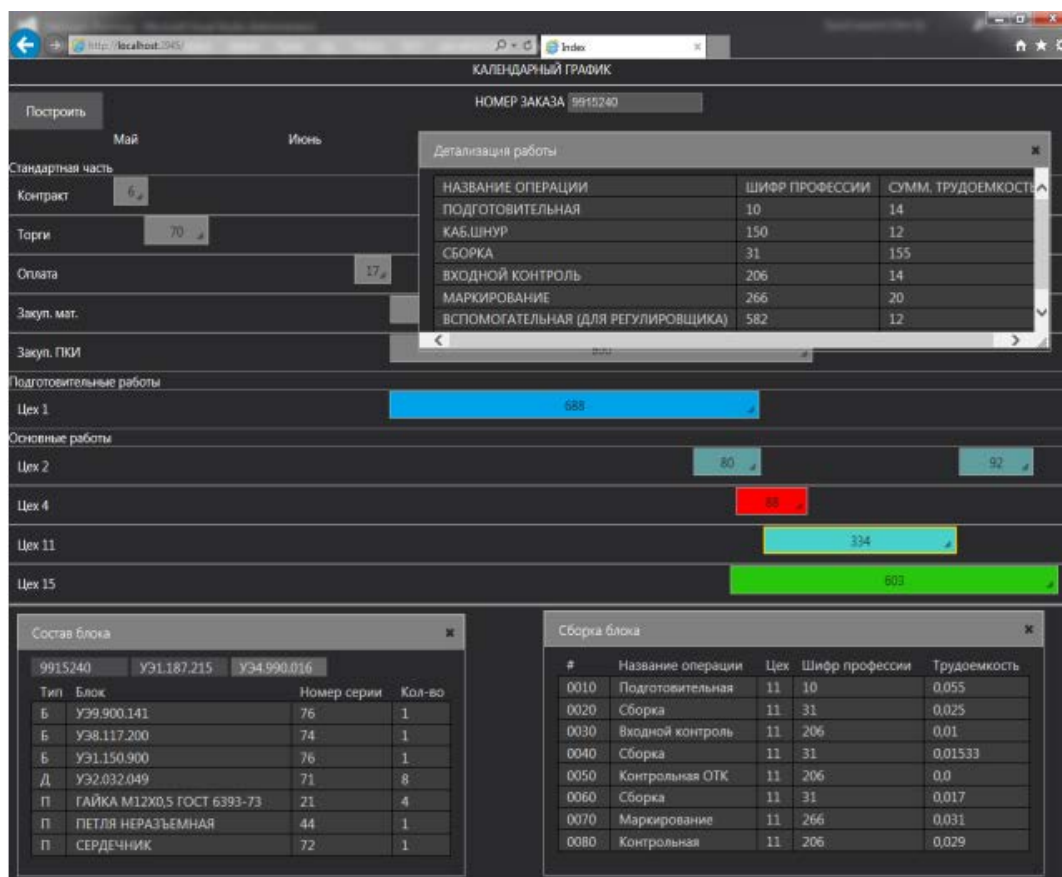


Рис.2 - Визуализация системы производственного сетевого планирования

Результатом данного проекта будет являться интегрированное программное решение сетевого производственного планирования. Сетевая модель визуально представляет собой набор полосок, состоящих из двух главных осей: задач и времени. Каждому временному промежутку приписывается определённые задачи или операции, которые должны быть выполнены. Следует отметить, что разработка данной системы ведется в кооперации с отделом автоматизированных систем управления предприятия АО «НПП «Радиосвязь». Результаты исследований планируется использовать в производственном процессе данного предприятия.

Список литературы

1. Оперативно-календарное планирование сборочных технологических процессов радиоэлектронных изделий / Е. Г. Есюкова, Е. Е. Носкова // Журнал Сибирского федерального университета. Техника и технологии. – 2014. – Т. 7 № 7. – С. 779–790.
2. Новицкий Н. И. Сетевое планирование и управление производством: Учеб.-практ. пособие / Н. И. Новицкий. – М.: Новое издание, 2004. – 159 с.

О МЕТОДАХ ГЕНЕРАЦИИ РАЗНООБРАЗИЯ АНСАМБЛЕЙ ПАРАЛЛЕЛЬНОГО ОБУЧЕНИЯ В ЗАДАЧАХ ВОССТАНОВЛЕНИЯ РЕГРЕССИИ

Мангалова Е.С.

научный руководитель канд. техн. наук Шестернева О.В.

Сибирский государственный аэрокосмический университет

им. ак. М.Ф. Решетнева

Фундаментальной задачей, возникающей при построении ансамблей параллельного обучения в задачах восстановления регрессии, является поддержание различия индивидуальных моделей (генерация разнообразия ансамбля) [1]. Объединение схожих моделей (пусть и очень точных) в ансамбле не может привести к значимому повышению точности восстановления зависимости.

Трудность в генерации разнообразия возникает из-за того, что при решении некоторой конкретной задачи индивидуальные модели обучаются по одной обучающей выборке и вследствие этого обычно сильно коррелированы. Из-за чего некоторые теоретически доказанные подходы к построению ансамбля становятся неэффективными при решении прикладных задач, поскольку основаны на предположении независимости индивидуальных моделей.

Также задача генерации разнообразия усложняется требованием к достаточной точности индивидуальных моделей. Отыскание компромисса между точностью индивидуальных моделей и их разнообразием является одной из ключевых задач при построении ансамбля моделей.

Зависимость ошибки ансамбля от разнообразия моделей в ансамбле. В работе [2] показано, что ошибка коллективной модели зависит от точности индивидуальных моделей и их разнообразия следующим образом:

$$err(H | \bar{x}) = \sum_{i=1}^N w_i err(h_i | \bar{x}) - ambi(H | \bar{x}),$$

в том случае, если правило объединения моделей представляет собой среднее взвешенное, а в качестве меры точности выбрана среднеквадратическая ошибка

$$err(h_i | \bar{x}) = (f(\bar{x}) - h_i(\bar{x}))^2,$$

$$err(H | \bar{x}) = (f(\bar{x}) - H(\bar{x}))^2,$$

неопределенность ансамбля в точке \bar{x} определяется как:

$$ambi(H | \bar{x}) = \sum_{i=1}^N w_i (h_i(\bar{x}) - H(\bar{x}))^2$$

и показывает рассогласование среди индивидуальных моделей в точке \bar{x} . Среднее значение рассогласования в точках выборки (валидационной или тестовой) будем понимать под мерой разнообразия ансамбля.

Эмпирическая оценка разнообразия и точности индивидуальных моделей. Для эмпирической оценки разнообразия и точности индивидуальных моделей будем использовать принцип кратной кросс-проверки:

1. Исходная выборка делится на обучающую D_{train} и экзаменующую выборки D_{test} .

2. Из обучающей выборки D_{train} формируются N обучающих подмножеств D_t , $t = 1, \dots, N$, которые участвуют в построении N индивидуальных моделей.

3. Обученные модели тестируются на выборке D_{test} : оцениваются разнообразие и точность индивидуальных моделей.

Процесс повторяется несколько раз с целью получения более точных эмпирических оценок разнообразия и точности индивидуальных моделей.

Соотношения между размерами подвыборок D_{train} и D_{test} , D_{train} и $D_t, t = 1, \dots, N$, зависит от размерности исходной выборки.

В случае малого объема исходной выборки может быть использован крайний случай кросс-проверки – скользящий экзамен (размерность D_{test} - одно наблюдение). Сложность в данном случае $O(n \times N)$, где x - сложность индивидуальной модели, для дерева регрессии - $O(n^2 \log n N)$, n – объем выборки. При больших объемах выборок применение скользящего экзамена становится недопустимо и требуется уменьшать соотношение между объемом обучающей выборки D_{train} и объемом тестовой выборки D_{test} .

Аналогичная ситуация происходит и с соотношением между объемом обучающей выборки D_{train} и объема проверочных выборок D_t .

Манипуляции с обучающими подмножествами. Манипуляция с обучающими подмножествами является самым распространенным и эффективным методом генерации разнообразия ансамбля параллельного обучения. Из обучающей выборки могут быть сформированы различные обучающие подмножества. Чем меньшие мощности имеют попарные пересечения этих подмножеств, тем более разнообразны обученные на них индивидуальные модели. Покажем влияние размерности обучающих подмножеств и правила их формирования на разнообразие ансамбля:

Из обучающей выборки D_{train} формируются N обучающих подмножеств $D_t, t = 1, \dots, N$, которые участвуют в построении N индивидуальных моделей. Формирование выборки D_t происходит таким образом, чтобы $|D_t| = |D_{train}|$, и D_t содержало $r|D_{train}|$ уникальных наблюдений из D_{train} , где r – параметр алгоритма генерации разнообразия.

На рисунке 1 показана зависимости между оценками разнообразия и параметром r , для задачи прогноза стоимость жилья в пригородах Бостона [3]. 13 входных переменные: уровень преступности на душу населения в пригороде, доля жилых земель, доля индустриальных земель, наличие реки в пригороде, концентрация оксида азота, среднее число комнат, доля жилых строений, построенных до 1940 г., взвешенное расстояние до 5 бизнес-центров Бостона, индекс доступности радикальных магистралей, уровень налогов, уровень образованности населения, этнический состав населения, процент бедного населения. Целевая переменная: средняя цена жилых домов в тысячах долларов. Количество наблюдений: 506. Индивидуальная модель – модель дерева регрессии.

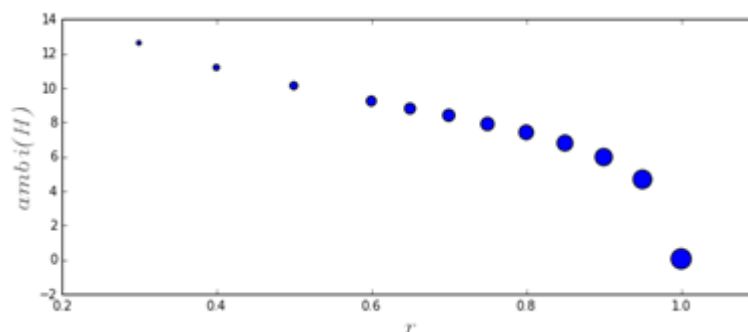


Рис.1 - Зависимость между оценками разнообразия и параметром r

Чем больше параметр алгоритма, тем более похожи между собой обучающие выборки и обученные на них модели. Соответственно, чем меньше данных мы

используем для построения моделей, тем они менее точные (рис. 2). Однако, объединение менее точных, но более разнообразных моделей в ансамбль позволяет повысить точность ансамбля (рис. 3).

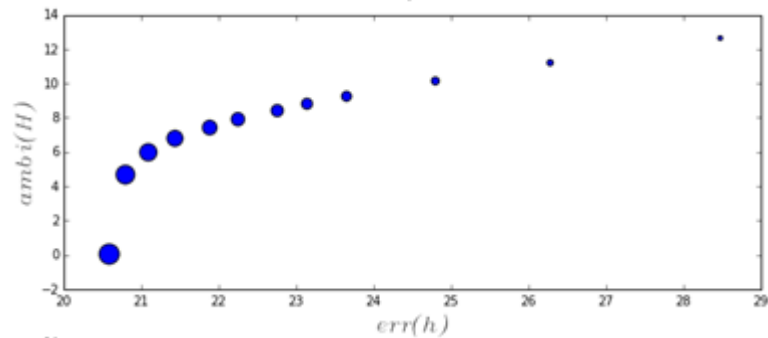


Рис.2 - Зависимость между оценками разнообразия и средней точности индивидуальных моделей

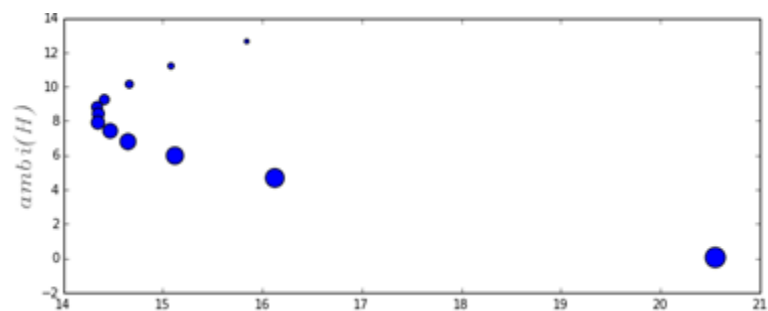


Рис.3 - Зависимость между оценками разнообразия и точности ансамбля

Манипуляции со входными переменными. Модели, входящие в ансамбль, могут отличаться друг от друга не только наблюдениями, на которых они обучены, но набором входных переменных. Зачастую выбор того или иного подпространства входных переменных при построении модели в ансамбле соответствуют различным взглядам на исходные данные.

Из исходного множества признаков формируются N подмножеств таким образом, что количество признаков в каждом равно r , где r – параметр алгоритма генерации разнообразия. По подмножествам признаков из обучающей выборки D_{train} строится N индивидуальных моделей, формирующих ансамбль.

Покажем работоспособность метода на той же задаче. Чем меньше признаков мы используем для построения моделей, тем они менее точные (рис. 4). Объединение менее точных, но более разнообразных моделей в ансамбль позволяет повысить точность ансамбля (рис. 5).

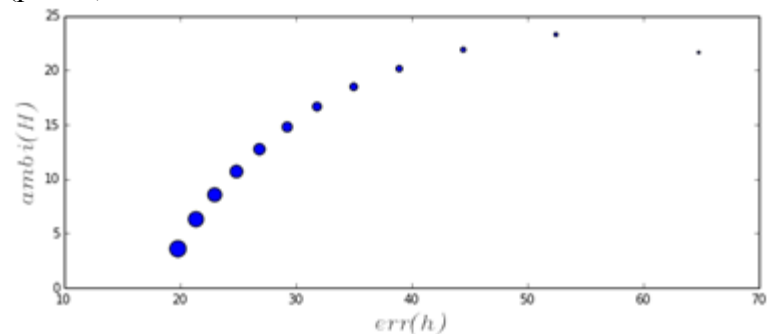


Рис.4 - Зависимость между оценками разнообразия и средней точности индивидуальных моделей

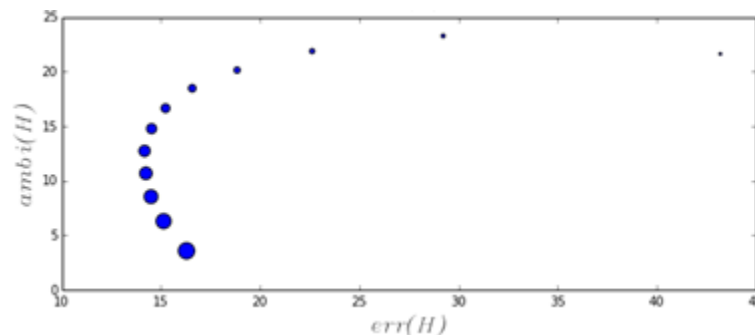


Рис.5 - Зависимость между оценками разнообразия и точности ансамбля

Манипуляции с параметрами алгоритмов обучения. Разнообразие индивидуальных моделей может достигаться с использованием различных параметров алгоритмов.

По обучающей выборке D_{train} строится N индивидуальных моделей с разными параметрами. Например, различные начальные веса нейронных сетей, различные критерии расщепления деревьев и т.д.

Однако, в ряде случаев стоит ограничивать степень различия параметров. Например, если при построении ансамбля деревьев регрессии варьируется размер минимального листа, то он не должен принимать значение от минимально возможного (1 наблюдение) до максимально возможного (n наблюдений). В таком случае в ансамбле будут содержаться индивидуальные модели, точность которых существенно разная. Как следствие, индивидуальные модели с существенно низшей точностью будут уменьшать точность всего ансамбля.

Если же параметры не влияют серьезно на точность индивидуальной модели или это влияние носит случайный характер (например, различные начальные веса нейронных сетей), то ограничение степени различия такие параметров алгоритма не требуется.

Следует заметить, что различные механизмы генерации разнообразия могут быть использованы совместно. Манипуляции с обучающими подмножествами, входными переменными и параметрами алгоритма могут быть применены как попарно, так и все одновременно.

Эффективность комбинации техник связана с тем, что рассмотренные техники позволяют достигать различных целей:

Манипуляции с обучающими подмножествами направлены на то, чтобы уменьшить влияние шума и аномальных значений выходной переменной (выбросов), снижая риск переобучения.

Манипуляции с входными переменными позволяют эффективнее обрабатывать данные в условиях малого объема выборок в сравнении с размерностью пространства признаков.

Список литературы

1. Zhou Z.-H. Ensemble Methods: Foundations and Algorithms. Machine Learning & Pattern Recognition series. Chapman & Hall/CRC, 2012. 236 p.
2. Krogh A., Vedelsby J. Neural network ensembles, cross validation and active learning // Advanced in Neural Information Processing System 7. Cambridge: MIT Press, 1995. P. 231-238.
3. Housing Data Set. UCI Machine Learning Repository: Data Sets. [Электронный ресурс]. URL: <https://archive.ics.uci.edu/ml/datasets/Housing> (дата обращения: 1.09.2015).

СИСТЕМА АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ ОБРАЗОВАТЕЛЬНЫМ ПРОЦЕССОМ С ОБРАТНОЙ СВЯЗЬЮ

Мартынов А.В.

научный руководитель д-р техн. наук, доц. Бронов С.А.

Институт космических и информационных технологий

Образовательный процесс представляет собой *образовательную программу* и её *реализацию*.

Процесс образования обычно представляется в качестве следующих последовательных действий (Рисунок 1).

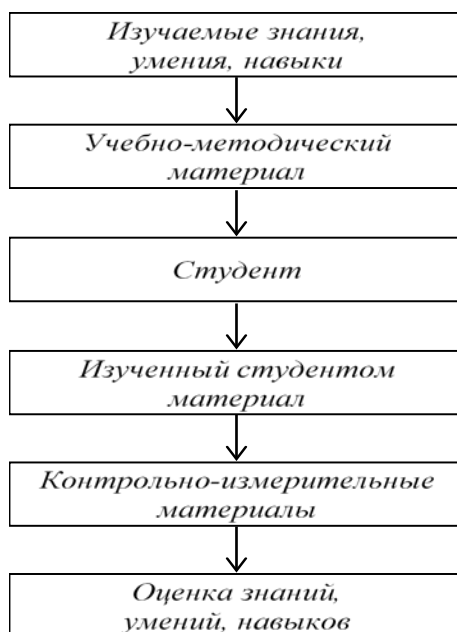


Рис.1 — Структурная схема процесса обучения

На входе процесса имеются *Изучаемые знания, умения и навыки*, из которых формируется *Учебно-методический материал* (пособия, презентации, лабораторные работы и т. п.). Эти материалы изучает *Студент*, в результате чего этот материал превращается в *Изученный студентом материал*, который содержит только те знания, умения и навыки, которые усвоены студентом. Чтобы выявить их, используются *Контрольно-измерительные материалы* (вопросы для экзамена, домашние задания, тесты и т. п.), с помощью которых выполняется *Оценка знаний, умений, навыков*.

Результатом рассмотренного процесса является принятие решения об успешности освоения учебного материала студентом с вытекающими из этого последствиями (оценка с зачёту, стипендия, пересдача, отчисление и т. д.).

С точки зрения системного анализа, данный процесс можно представить в виде обобщенной структурной схемы системы автоматического управления с обратной связью (Рисунок 2).

Система (Рисунок 2), вне контекста образовательного процесса, работает следующим образом.

Имеется *Объект управления*, которым данная система управляет.

Управлять — означает, что выходная величина *Объекта управления* должна принимать только те значения, которые считаются желаемыми.

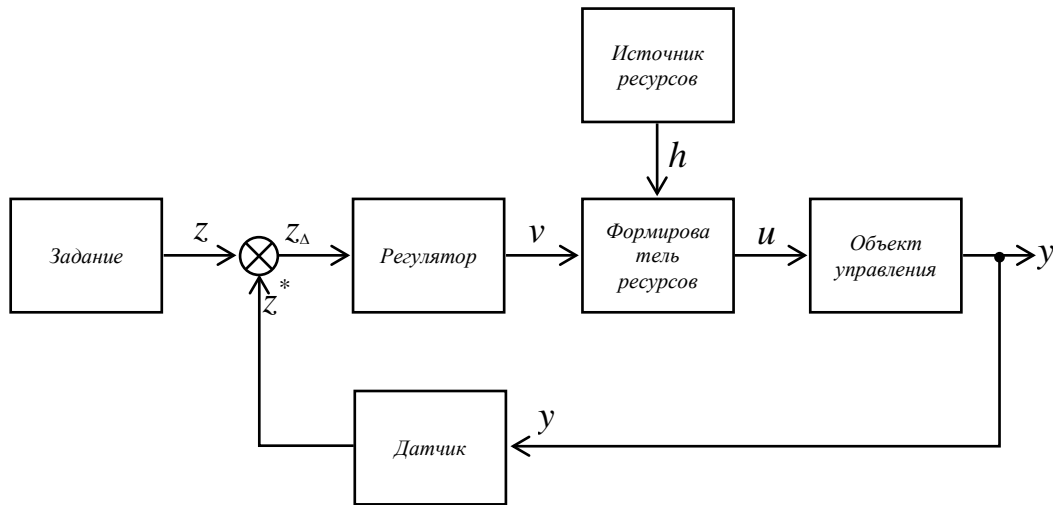


Рис.2 – Обобщенная структурная схема системы автоматического управления с обратной связью

С помощью блока *Задание* задаётся желаемое значение z выходной величины y . Система должна так сработать, чтобы значение выходной величины равнялось желаемому значению: $y = z$. Не всегда можно достичь точного равенства, тогда значение выходной величины y должно, по крайней мере, отличаться от желаемого z не более чем на некоторую ошибку:

$$y = z + z_{\Delta},$$

где значение ошибки z_x должна быть не больше максимально допустимого значения:

$$z_{\Delta} \leq z_{\Delta.\max}$$

Чтобы обеспечить

$$y = z + z_{\Delta} \text{ или } y = z + z_{\Delta} \text{ при } z_{\Delta} \leq z_{\Delta.\max}$$

выходная величина измеряется с помощью блока *Датчик*, на выходе которого формируется величина z^* , которая должна быть равна выходной величине:

$$z^* = y$$

Желаемая величина z сравнивается с измеренной выходной величиной z^* , определяется рассогласование между выходной величиной и желаемой и вычисляется *отклонение* ("ошибка", "невязка"):

$$z_{\Delta} = z - z^*$$

Этот сигнал ошибки проходит через блок *Регулятор*, который формирует на выходе новую величину

$$v = f_1(z_{\Delta}).$$

Эта величина поступает на вход блока *Формирователь ресурсов*, который связан с выше расположенным блоком *Источник ресурсов*. Эти блоки выполняют следующие функции.

Блок *Источник ресурсов* содержит материальные и нематериальные ресурсы, которые используются для приведения *Объекта управления* в требуемое состояние.

Блок *Формирователь ресурсов* осуществляет дозированную подачу ресурсов из *Источника ресурсов* в *Объект управления*, т. е. задаёт, какие именно ресурсы, в каком количестве и с каким темпом поступают в *Объект управления*.

Это формирование осуществляется под воздействием выходного сигнала блока *Регулятор*. Т. е. величина u — ресурс, изменяющийся во времени заданным образом; v — сигнал на изменение ресурса. Т. е. физическая природа величин v и u в общем случае различная: v — это сигнал, а u — это ресурс. В правильно действующей системе законы изменения во времени обеих величин должны быть одинаковыми.

Закон изменения во времени величины v определяет *Регулятор*, который делает это на основе анализа своего входного сигнала z_{Δ} , являющегося рассогласованием между реальным и желаемым значениями выходной величины: $z_{\Delta} = z - z^*$.

Если рассогласование отсутствует, то сигнал v остаётся неизменным (так как система работает правильно). Если рассогласование есть, то *Регулятор* изменяет сигнал v таким образом, чтобы выходной сигнал u стал равен заданному, т. е. чтобы ошибка $z_{\Delta} = z - z^* = 0$.

Здесь основная проблема заключается в том, чтобы выбрать правильный алгоритм работы *Регулятора*. Процедура определения этого алгоритма называется синтезом регулятора. При этом считается, что все остальные элементы системы (*Объект управления*, *Источник ресурсов*, *Формирователь ресурсов*, *Датчик*) уже заданы и изменению не подлежат — они в совокупности называются неизменной частью системы. Регулятор, структура и параметры которого синтезируются (определяются), называется изменяемой частью системы.

Применительно к образовательному процессу (Рисунок 2) блоки имеют следующий смысл.

Блок *Задание* включает в себя материал для обучения.

Блок *Объект управления* представляет собой студентов.

Блок *Источник ресурсов* содержит комплекс дидактических методов — способов, приёмов подачи материала.

Блок *Формирователь ресурсов* выбирает из источника ресурсов те дидактические методы, которые наиболее подходят для передачи материалов студентам.

Блок *Регулятор* формирует указания на выбор тех или иных дидактических методов подачи материала.

Блок *Датчик* представляет собой фонд оценочных средств, в частности, набор тестов.

Задачей системы является формирование у студентов компетенций (знаний, умений и навыков) путём преподавания им учебного материала в той или иной форме теми или иными методами. Формы и методы выбираются преподавателями с учётом личного опыта, методических рекомендаций и т. п. условий.

Система работает следующим образом.

На выходе блока *объект управления* присутствуют дидактические единицы, усвоенные студентами. Такая структура обладает некоторыми общими системными свойствами, как всякая структура систем с обратной связью.

Обратная связь в системах, в общем случае, может быть положительной и отрицательной. В технических объектах, которые описываются дифференциальными и алгебраическими уравнениями, это реализуется непосредственно в блоке определения

рассогласования (сигнал обратной связи вычитается из желаемого или суммируется с ним). Для уменьшения рассогласования необходимо использовать отрицательную обратную связь.

Но в соответствии с теорией систем при этом может возникнуть неустойчивость. Поэтому при введении обратной связи первая проблема — обеспечение устойчивости системы и лишь во вторую очередь — повышение качества.

В организационных объектах используется только отрицательная обратная связь: всегда стараются ввести такую коррекцию, чтобы уменьшить рассогласование между желаемым и полученным результатами. Но в действительности, возможно неправильное понимание того, какие корректирующие действия имеют смысл положительной или отрицательной связи. Применительно к образовательному процессу это необходимо рассмотреть отдельно. Например, если в результате тестирования выявлено, что какой-то материал студентом не усвоен, преподаватель, корректирующий учебный материал, может добавить разъяснения (которые, как он думает, повысят степень понимания), которые только увеличат время, необходимое для изучения. Тогда студент тем более не сможет осилить весь материал, и тем самым процесс станет неустойчивым. В ином случае преподаватель может уменьшить разнообразие материала и углублённо рассмотреть отдельные разделы — тогда обратная связь может оказаться отрицательной и способствовать повышению качества усвоения материала.

Учебный материал является, с одной стороны, материальным носителем управляемых характеристик. С другой стороны, его организация является сигналом, обеспечивающим подачу этого материала.

Материал может обладать качественными характеристиками: ясность подачи материала, понятность и количественными параметрами: объём материала и интенсивность (скорость) его подачи. Все качественные характеристики должны получить количественное наполнение: например, ясность изложения может представляться как логическая последовательность, отсутствие её нарушений; использование только тех базовых знаний, которые уже имеются у студента и др. Неустойчивость материала возникает только в динамических системах, т. е. когда процессы развиваются во времени.

Возможные варианты коррекции учебного материала:

- 1) изменение объёма материала;
- 2) изменение интенсивности подачи материала;
- 3) изменение последовательности подачи материала;
- 4) изменение формы подачи материала (например, вместо лекций — практические занятия, вместо лабораторных работ — домашние задания, вместо текстов — мультимедийные материалы в виде слайдов, анимированных представлений, видео, звука и т. д.).

Таким образом, была рассмотрена система автоматического управления образовательным процессом, и поставлен ряд задач, и проблем, которые будут решены в ходе дальнейшего исследования, а также определен вектор направления исследования.

Список литературы

1. Бронев С.А. Методологические проблемы автоматизированного формирования образовательных программ в рамках ФГОС ВПО
2. Антонов, А.В. Системный анализ: Учебник для вузов / А.В. Антонов. - М.: Высш. шк., 2008. - 454 с.



КОНЦЕПЦИЯ УПРАВЛЕНИЯ ИНДУКТОРНЫМ ДВИГАТЕЛЕМ ДВОЙНОГО ПИТАНИЯ ДЛЯ ЭЛЕКТРОПРИВОДОВ СИСТЕМ ПОВОРОТА КОСМИЧЕСКИХ АППАРАТОВ

Никитин Р. М.

научный руководитель д-р техн. наук, проф. Бронов С. А.

Сибирский федеральный университет

При проектировании систем управления различного назначения, инженеры-исследователи как правило, стремятся минимизировать использование различных датчиков, путем расчета необходимых параметров с использованием различных моделей проектируемого объекта. В основном к датчикам применяются следующие основные требования: высокая точность, надежность и помехоустойчивость. Расчет необходимых параметров с помощью модели в целом надежен и не имеет проблем с помехоустойчивостью. Однако для достижения необходимой точности, особенно для параметров систем реального времени, для расчета в модели требуется большие вычислительные мощности.

В отрасли производства космических аппаратов (КА), применение большого количества датчиков в электроприводных системах КА, существенно усложняет систему управления электроприводом и снижает ее надежность. Для регулирования антенн и солнечных батарей КА обычно используются электроприводы с синхронными двигателями с постоянными магнитами (СДПМ) и синхронными двигателями с электромагнитной редукцией (СДЭР), а также предлагают к использованию индукторные двигатели двойного питания (ИДДП)^[1]. При построении систем управления представленных двигателей, предлагается использовать датчики угла поворота, положения, скорости ротора двигателя (датчики Холла, механические, оптические и др.)^[1]. В случае с СДПМ и СДЭР, для построения эффективной системы управления электроприводов с такими типами двигателей, необходимо создать замкнутую систему с обратной связью, что подразумевает наличие хотя бы одного датчика положения ротора двигателя. Использование ИДДП при проектировании электроприводов, позволяет создавать как замкнутые, так и разомкнутые системы управления электроприводом^[2].

Угол поворота ротора ИДДП будет основной регулируемой величиной системе управления электроприводом. Он определяется следующим выражением^[3]:

$$\theta_r = \frac{\theta_1 - \theta_2}{Z_r},$$

где θ_r — угол поворота вала двигателя; θ_1 , θ_2 — текущие фазовые сдвиги первого и второго питающих напряжений; Z_r — коэффициент электромагнитной редукции.

Но в динамических режимах работы электропривода необходимо контролировать также другие переменные состояния, в частности, скорость и угол нагрузки. Угол нагрузки θ_M невозможно измерить каким-либо датчиком непосредственно, но его можно определить с использованием текущих фаз питающих напряжений и угла поворота^[3]:

$$\theta_M = \theta_1 - \theta_2 - Z_r \theta_r$$

Таким образом, для измерения угла нагрузки достаточно измерить угол поворота вала двигателя, а также иметь информацию о текущих фазах питающих напряжений.

Имея информацию о выше перечисленных параметрах, с помощью модели ИДДП можно вычислить другие необходимые для регулирования привода параметры (токи, потокосцепления, угловая скорость и пр.). В результате система управления электроприводом будет выглядеть следующим образом (рис.1)^[3]:

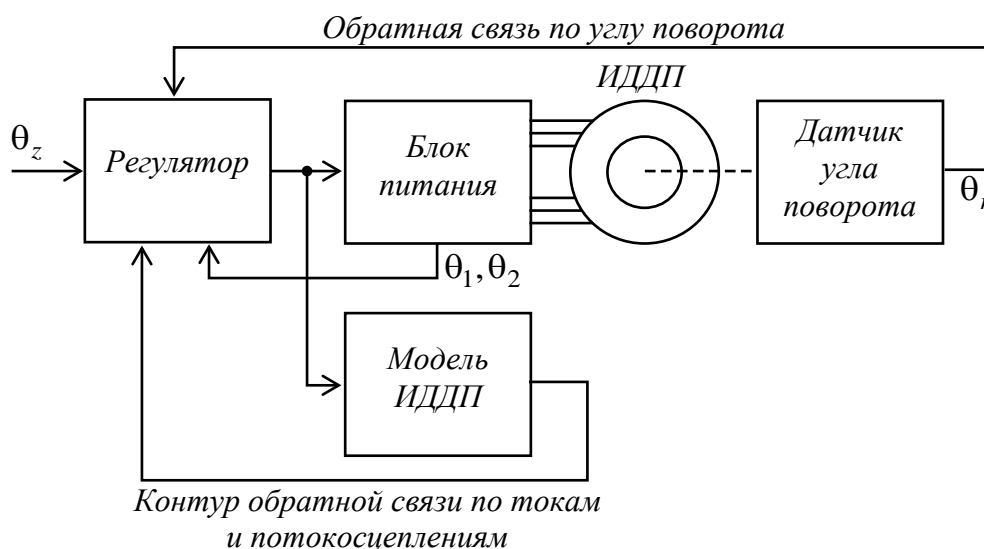


Рис.1 – управление ИДДП с моделью и датчиком угла поворота

Поскольку основной регулируемой переменной является угол поворота, то необходимо использовать датчик угла поворота, который дает достоверное значение угла и способствует исключению погрешностей, возникающих при расчете в модели. Этот факт означает что полностью датчико-независимой системы управления электроприводом построить не получится. Однако, в разрабатываемой концепции, предлагается ввести функциональную зависимость угла поворота θ_r от параметров вращаемого объекта $x_0...x_n$:

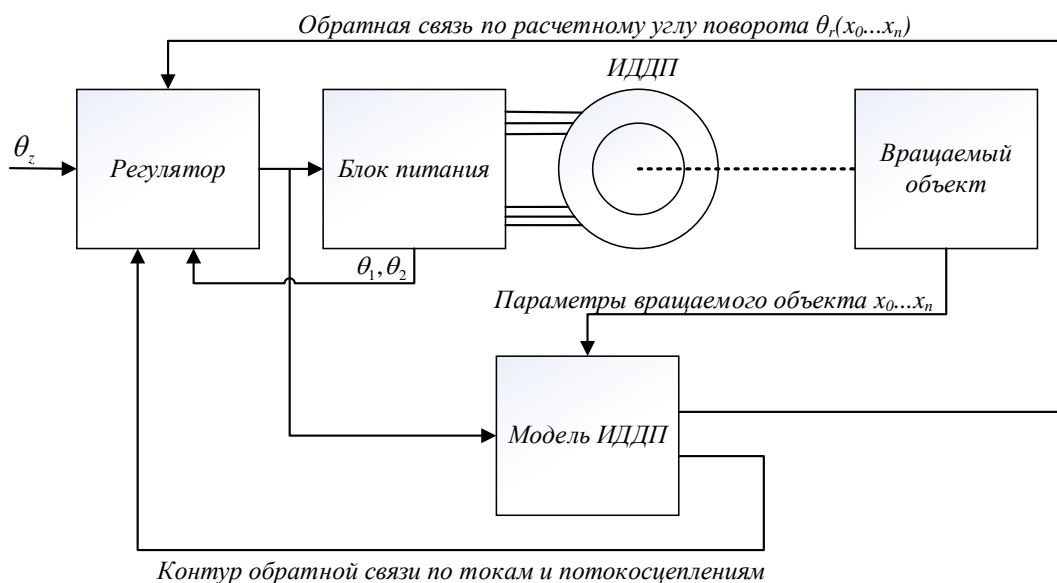


Рис.2 – управление ИДДП с моделью и расчётным углом поворота

Таким образом предлагает рассчитывать угол поворота ротора в модели ИДДП при помощи детектируемых параметров вращаемого объекта. Параметры объекта предлагается выбирать исходя из его назначения. К примеру, для электроприводной системы солнечной батареи КА одним из таких параметров может быть напряжение, возникающие при поглощении солнечной энергии, за определённый промежуток времени. Для электроприводной системы антенны КА это может быть уровень сигнала, качество сигнала, уровень помех и т.п.

В целом, такой подход к управлению электроприводной системой КА с ИДДП, усложнит вычисления в модели ИДДП, что потребует дополнительных вычислительных мощностей для управления приводом. Однако даже в этом случае, количество необходимых вычислений будет несоизмеримо мало по сравнению с количеством вычислений в модели для полностью бездатчиковой реализации с требуемой точностью.

Список литературы

1. Бронов, С. А. Проектирование электроприводов систем поворота антенн и батарей солнечных космических аппаратов / С. А. Бронов, П. В. Авласко, В. А. Марарескул, В. А. Поваляев // Авиакосмическое приборостроение. — 2010. — № 2. — С. 1—6.

2. Бронов, С. А. Управление индукторными двигателями двойного питания в позиционных электроприводах космических аппаратов / С. А. Бронов, П. В. Авласко, А. В. Марарескул, В. А. Поваляев, В. П. Усов, В. С. Куповых // Авиакосмическое приборостроение. — 2010. — № 9. — С. 34—40.

3. Бронов, С. А. Прецизионные электроприводы космических аппаратов для дистанционного зондирования Земли / С.А. Бронов, Н. А. Никулин, П. В. Авласко, В. А. Поваляев, Р. М. Никитин, М. Н. Ульяновченко // Региональные проблемы дистанционного зондирования Земли : материалы II Междунар. науч. конференция, 22–25 сентября 2015, г. Красноярск / науч. ред. Е. А. Ваганов; отв. ред. М. В. Носков. — Красноярск : Сиб. федер. ун-т, 2015. — С. 43—45. — ISBN 978-5-7638-3306



РАЗРАБОТКА POINT – OF INTEREST РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ ПРИ ИСПОЛЬЗОВАНИИ LBSN СОЦИАЛЬНЫХ СЕТЕЙ

Пахомова К.И.

научный руководитель д-р физ-мат наук Добранец Б.С.

Сибирский федеральный университет

Жизнь современного человека весьма динамична, каждый день мы планируем и совершаем определённые действия, такие как (поход в магазин, работа, поездка на учебу, встречи с друзьями в кафе, спортивные залы и т.д.). Каждый новый день человека подобен циклу, а именно совершаемые процессы происходят последовательно одно за другим. Для того, чтобы организовать и дать точные рекомендации, необходимо определить эти циклы и извлечь именно интересную информацию из LBSN (Location Based Social Network). На основе полученных результатов и применения алгоритмов рекомендательных систем разработать рекомендательную систему жизненной модели человека.

Современные LBSMs позволяют распространение пользовательского и мультимедийного контента, обмен сообщениями, определение текущей геолокации, с их помощью можно анализировать как собственную жизнь, так и жизнь, интересующих вас людей. Обычно, привычки людей не претерпевают изменений и человек сам того не осознавая существует по созданным им самим правилам, например, человек начинает свой день с чашки кофе в любимом ему баре, затем он отправляется на работу (учебу), обедает в столовой, а вечером посещает любимый фитнес клуб. Практически у каждого человека существует аналогичная расписание дня, а именно человек живет по определенной подсознательно созданной жизненной модели, зависящей от собственных привычек и образа жизни, а также происходящей в определенной локации. В случаях, когда человек меняет локацию, например путешествие, отпуск, поездка к родственникам или друзьям заставляют отходить от рутинных правил, но при этом привычки не претерпевают изменений. Цель данной работы – это разработать систему, которая бы помогала человеку адаптироваться за короткий промежуток времени, при этом не изменяя свои привычки и вкусы. Исследовательская работа состоит из двух основных стадий, а именно определение жизненной модели человека и разработка рекомендательной системы, основываясь на постоянной жизненной модели.

Каждый человек имеет собственные привычки, пристрастия и личные интересы. Темп современной жизни настолько быстр, что человек постоянно должен планировать свое личное время, для того чтобы выполнить все намеченные дела. Темп жизни человека заставляет его находиться в постоянном цикле, подчиняясь собственным – личным правилам. На основании этих правил человек моделирует свое поведение и осуществляет ежедневное планирование. Вследствие этого предложим определение жизненной модели человека, а именно обратимся к теории графов.

Определение Модель человеческой жизни $G = (I, A)$ – это ориентированный граф, где $U = \{u_1, u_2, \dots, u_n\}$ – множество пользователей и $I = \{i_1, i_2, \dots, i_m\}$ множество point- of- interest (точек интереса) POI, где m, n – это количество пользователей и интересов, в которых $POI = \{UserID, Time, Location(x, y), Topic\}$ представлено как опеределенный пользователь $UserId$, в определенное время t и в конкретной точке location (x, y) оставляет комментарий в интересном topic для него месте.

Для построения человеческой модели были взяты данные из LBSN Foursquare,

который в настоящее время является одним из самых популярных социальных сетей, с ориентацией на локацию пользователя, в России данная социальная сеть действительна с 2010 года. Итак, анализируя данные с Foursquare, и ссылаясь на выше упомянутое определение, получаем граф жизненной модели человека, рисунок 1.



Рис.1 – Жизненная модель человека Июнь 2010

На данном рисунке представлен жизненный цикл человека, где узлами являются интересы и ежедневные правила пользователя, именуемые как POI (point of interest), а ребра показывают последовательность посещаемых мест. Анализируя этот граф, мы можем изучать личные интересы человека и выявить самые распространенные, производить классификацию над ними, а также сопоставлять и оценивать их. Пользователь под номером 47 имеет 566 отмеченных им локаций, 27 июня 2010 года 9 отмеченных локаций составило 1.62% из всего количества. Жизненная модель данного пользователя началась с 1:05:45 часов и этому времени соответствует POI – Nightlife (ночная жизнь).

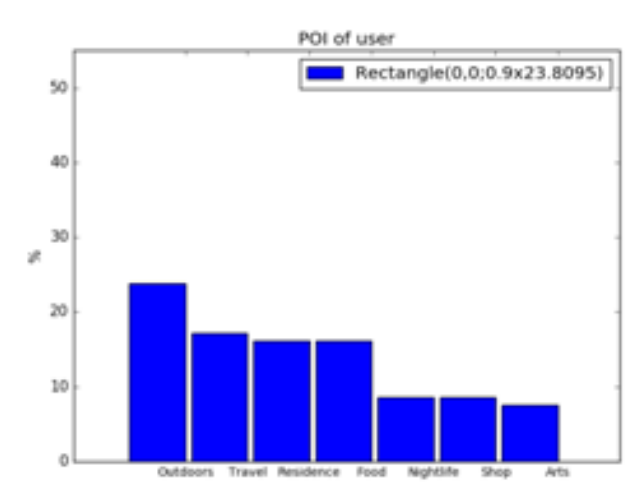


Рис.2 - POI пользователя в %

После 4:37:39 пользователь отметил POI как Residence (дом), которая имеет специальные координаты или Id локации (4b5908fef964a5207a7928e3) и собственное название (FIPS HQ). Далее в 14:08:15 он пошел в магазин, в 15:02:49 Prospect Perk Café, далее POI Food и день пользователя заканчивается в 19:57:02 Voxers NYC Sports bar. В этот момент его повседневный жизненный цикл замыкается. Размер узла графа отражает количество отмеченных POI, на рисунке 1 наблюдается градация POI от меньшего

размера к большому, другими словами от часто посещаемого интереса, к менее посещаемую, например, в Июне 27 POI Shop соответствует 3 отметки – диаметр узла увеличенный, а POI Outdoors соответствует 1 отметка. Анализ полученного графа модели жизненного цикла человека осуществляется по средствам статистических моделей, таких как degree distribution (распределение узлов по числу связей), modularity class (классы модулярности), PageRank distribution (PageRank распределение). Каждый из данных способ доказывает существование жизненной модели человека и помогает решить проблему рекомендации, по средствам определения параметров. Так, модулярность разбивает граф рисунок 1 на два класса, в одном сконцентрированы узлы: Shop, Nightlife, Food; во втором: Outdoors, Residence.

Персонализированные POI рекомендация - данная задача подразумевает под собой следующую цель, а именно помочь пользователям отфильтровать неинтересные POI и сократить время принятия решений, которые могли бы также приносить пользу не только самому пользователю, но и осуществлять виртуальную функцию.

В LBSN, происходят физические взаимодействия между пользователями и объектами инфраструктуры, что является уникальным отличительным свойством POI от традиционных рекомендательных систем. По средствам этого POI рекомендательные системы совмещают в себе определенный перечень дополнительной информации, такой как географическое положение, временная зависимость, периодичность посещения определенных локаций, а также заинтересованность определенных локаций. Данные факторы в совокупности расширяют масштабность рекомендаций и делают их более интересными как для самого пользователя, так и для социума, в общем. В частности, географическое положение играет самую важную роль в POI рекомендации. На текущий день существует не малое количество рекомендательных алгоритмов, но данные алгоритмы узкоспециализированны, определенные оперируют с временными или хронологическими данными, следующие направлены на определение локации (месторасположения), также существуют алгоритмы, работающие с данными, определяющими отношения пользователей. Данная работа ориентирована на разработку оптимального алгоритма, синтезирующего в себе воедино все вышеперечисленные факторы для предоставления качественной рекомендации за короткий промежуток времени.

Список литературы

- 1) Y. Yu, X. Chen, A Survey of Point-of-Interest Recommendation in Local-Based Social Networks. Association for the Advancement of Artificial Intelligence, 2015.
- 2) H. Gao. Personalized POI Recommendation on Location-Based Social Networks. Arizona state university, 2014.
- 3) H. Gao, J. Tang, X. Hu, H. Liu. Content-Aware Point of Interest Recommendation on Location-Based Social Networks. Association for the Advancement of Artificial Intelligence, 2015.
- 4) Y. Yu, X. Chen, A Survey of Point-of-Interest Recommendation in Local-Based Social Networks. Association for the Advancement of Artificial Intelligence, 2015.
- 5) Huiji Gao. Personalized POI Recommendation on Location-Based Social Networks. Arizona State University, 2014.



ИСПОЛЬЗОВАНИЕ СЕМАНТИЧЕСКОЙ МИКРОРАЗМЕТКИ ПРИ ФОРМИРОВАНИИ НЕСТРУКТУРИРОВАННОГО КОНТЕНТА

Печерица А.Т.

Томский институт систем управления и радиоэлектроники

В настоящее время имеется огромное количество информационных порталов и блогов, где пользователи самостоятельно генерируют контент. В результате анализа наиболее популярных блогов было выявлено, что информацию в них принято структурировать с использованием тегов, хэштегов, тематических категорий. На самом деле использование такого рода структуризации не всегда удобно, поскольку при таких обстоятельствах сам контент не несет информации, необходимой поисковым системам. Следовательно, и пользователям найти информацию, если она не помечена тэгами становится сложнее. Наилучшим решением в структуризации информации становится использование формата микроразметки [1].

Удобство использования микроразметки было выявлено в рамках создания информационного портала, когда встала задача структуризации информации. Под микроразметкой (или семантической разметкой) подразумевается разметка страницы с дополнительными атрибутами, которые указывают поисковым роботам на непосредственное содержание страницы. Цель семантической разметки – сделать интернет более понятным и структурированным, облегчить поисковым системам и специальным программам извлечение и обработку информации для удобного её представления в результатах поиска. Важно отметить, что разметка встраивается в HTML-коде страниц и не требует создания отдельных экспортных файлов [1-2].

Позволить пользователям максимально удобно создавать посты, не задумываясь о категоризации публикуемого контента. Для структуризации информации использовать микроформаты, таким образом обеспечивая связь с различными заведениями, мероприятиями и т.п. Отличным решением данной задачи является использование словарей микроданных, в частности schema.org.

Schema.org предоставляет общедоступный словарь, с помощью которого вебмастера могут разметать страницы так, чтобы они были понятны самым распространенным поисковым системам: Google, Yahoo, Bing и Яндекс, которые в свою очередь принимают участие в разработке этого словаря [3]. Таким образом, основным преимуществом Schema.org, становится ее взаимодействие с поисковыми системами, посредством которого они корректно воспринимают всю, добавленную пользователем, информацию.

Результатом изучения всех возможностей использования семантической микроразметки и словаря schema.org стал прототип редактора постов, который в дальнейшем будет использован на информационном портале. Разработанный прототип редактора постов отличается от большинства своих аналогов наличием встроенной микроразметки, сопровождающей любой новый пост (рис.1). Удобство данного прототипа заключается в отсутствии лишних требований к пользователю. Преимущество – максимальная структуризация контента.

Новый пост

Событие


Заголовок поста

Название события

Описание	+
Организатор	+
Участники	+
Продолжительность	+
Изображение	+
Видео	+
Дата	+
Место	+

С чем связано?

<input type="checkbox"/> Бизнес	<input type="checkbox"/> Развлечения
<input type="checkbox"/> Дети	<input type="checkbox"/> Социальное
<input type="checkbox"/> Литература	<input type="checkbox"/> Спорт
<input type="checkbox"/> Музыка	<input type="checkbox"/> Танцы
<input type="checkbox"/> Образование	<input type="checkbox"/> Фестиваль



Текст поста

Добавить

Рис.1 - Внешний вид генератора постов

В результате использования данного редактора постов, от пользователя требуется минимальное количество действий: выбрать категорию желаемого поста и заполнить несколько полей, относящихся к данной категории. Стоит отметить, что жестких ограничений к заполнению определенных полей не предъявляется, а вся необходимая микроразметка добавляется при этом системой автоматически. Итогом работы становится читабельный и аккуратно оформленный контент для пользователя и понятная информация для поисковых систем. Наглядно ознакомиться с примером неразмеченного контента (без использования оформления) можно ниже (рис. 2).

Танцевальный флешмоб в День томича

9 сентября в 2 часа дня концертную программу Дня томича на Новособорной площади открыл массовый танцевальный флешмоб под песню «Томск – это все мы!», где участвовали студенты и школьники, к которым присоединились и все желающие.



Рис.2 - Пример неразмеченного контента

На рисунке 2 достаточно хорошо видно наличие заголовка, текста и видео, однако для поисковых систем этот текст не несет никакой информации, так как в процессе создания поста не было добавлены какие-либо теги или тематические категории. Контент с наличием разметки значительно отличается от неразмеченного (рис. 3).

Танцевальный флешмоб в День томича
Концертную программу Дня томича на Новособорной площади открыл массовый танцевальный флешмоб под песню «Томск – это все мы!», где участвовали студенты и школьники, к которым присоединились и все желающие.
Starts: 09/25/2015 2:00PM
Duration: 00:20
Площадь
Новособорная

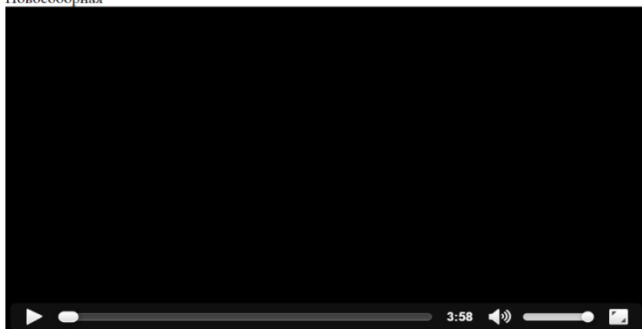


Рис.3 - Пример размеченного контента

Размеченный контент, хотя и имеет тот же заголовок и описание, все же отличается наличием заранее заготовленных полей, таких как время начала, продолжительность и место, которые вынесены отдельно для наиболее комфортного восприятия информации. Неизменным осталось прикрепленное к посту видео. Вместе со структуризацией информации, контент стал понятен поисковым системам, которые умеют различать время начала мероприятия, его продолжительность и место проведения, а также само название определенного события. Естественно, такую информацию найти будет гораздо проще, используя специальные фильтры (например, события, которые произошли 9 сентября днем).

В результате изучения понятия семантической микроразметки и рассмотрения примеров её использования, в рамках группового-проектного обучения, было принято решение о дальнейшем ее внедрении в процесс разработки информационного портала. Семантическая микроразметка станет незаменимым механизмом в процессе формирования структурированного контента.

Список литературы

1. Яндекс.Вебмастер.Введение в schema.org [Электронный ресурс]. – Режим доступа: <https://yandex.ru/support/webmaster/schema-org/intro-schema-org.xml>, свободный (дата обращения: 25.09.2015).
2. Хабрхабр [Электронный ресурс]. – Режим доступа: <http://habrahabr.ru/company/yandex/blog/221881/>, свободный (дата обращения: 25.09.2015).
3. Schema.org. Get Started [Электронный ресурс]. – Режим доступа: <https://schema.org/docs/gs.html>, свободный (дата обращения: 25.09.2015).

О СПОСОБЕ ОЦЕНКИ ФОРМИРОВАНИЯ ПРОФЕССИОНАЛЬНЫХ КОМПЕТЕНЦИЙ

Позолотина Н.Б.

научный руководитель канд. техн. наук Якунин Ю.Ю.

Сибирский федеральный университет

Экономические, политические и социальные перемены в современном обществе требуют принципиально нового подхода к высшему образованию. Последние федеральные государственные образовательные стандарты в соответствии с принципами Болонского процесса ориентируются на выработку у студентов довольно значительного числа компетенций – динамического набора знаний, умений, навыков, моделей поведения, личностных и профессиональных качеств, которые позволят выпускнику стать конкурентоспособным на рынке труда и успешно реализовываться. Поэтому одной из главных задач ВУЗа является разработка образовательных программ (ОП) с четко обозначенными результатами обучения – компетенциями, соответствующими ФГОС ВО по конкретному направлению подготовки, а так же запросам потенциального работодателя.

Можно ли научить компетенциям? Отечественные и большинство зарубежных исследователей полагают, что компетенции динамичны, поскольку они не являются неизменным качеством в структуре личности человека, а способны развиваться, совершенствоваться или полностью исчезать при отсутствии стимула к их проявлению. Поэтому можно говорить об уровне сформированности компетенций, его мониторинге.

Следовательно, перед преподавателями, реализующими ОП, встает ряд сложных задач:

- представлять, какие результаты обучения должны быть получены в рамках конкретной дисциплины;
- четко определять типы и формы учебной деятельности, а так же современные образовательные технологии, которые лучше всего подходят для достижения запланированных результатов обучения;
- хорошо представлять средний объем времени, необходимый для выполнения каждого вида учебной деятельности;
- разработка оценочных средств, которые позволят проводить объективную комплексную оценку сформированных компетенций будущих специалистов.

ФГОС ВО предусматривает три вида компетенций: общекультурные (ОК), общепрофессиональные (ОПК) и профессиональные (ПК).

ОК формируются на основе личностных качеств обучаемых, поэтому оценка уровня сформированности компетенций может быть проведена методами социологических и психологических исследований (анкетирование, опрос, тест и т.д.). [1]

В связи с тем, что сама формулировка компетенций типа ОПК и ПК, прописанная в ФГОС ВО, затрудняет непосредственную оценку уровня их сформированности, требуется провести их декомпозицию и создать паспорта компетенций, выделив компоненты в виде результатов обучения.

В данном контексте паспорт компетенции – это документ, из которого преподаватель сможет получить систематизированную информацию о значимости компетенции для выпускника данной образовательной программы, ее структуре, этапах, возможных уровнях формирования.

Для примера рассмотрим профессиональную компетенцию ОП подготовки магистров направления 27.04.03 «Системный анализ и управление»: способность принимать непосредственное участие в учебной работе кафедр и других учебных подразделений организаций по направлению подготовки данному направлению подготовки (ПК-7). Выделим компоненты, соответствующие разным видам учебной работы преподавателей кафедры см. Рисунок 1.

Способность принимать непосредственное участие в учебной работе кафедр и других учебных подразделений организаций по направлению подготовки данному направлению подготовки (ПК-7)



Рис.1 - Пример декомпозиции

На основе выделенных компонент и перечня дисциплин, реализующих данную компетенцию, получим паспорт ПК-7:

Таблица 1 – Паспорт компетенции ПК-7

Компоненты компетенции	Дисциплины, практики, НИР	Этапы формирования компетенций (семестр)	Виды занятий	Оценочные средства
Разработка оригинальных лабораторных работ и практических заданий	Педагогическая практика	2	Практ. зан.	Отчет о выполнении
	Методическое обеспечение учебных дисциплин	2	Практ. зан.	Отчет о выполнении
Разработка учебно-методического обеспечения дисциплин	Методическое обеспечение учебных дисциплин	2	Практ. зан.	Отчет о выполнении
Разработка ЭОР	Электронные образовательные ресурсы	2	Практ. зан.	Отчет о выполнении
Педагогическая деятельность	Педагогическая практика	2	Лаб. зан. и/или практ. зан.	Отчет о выполнении
Учебное планирование	Методология учебного планирования	2	Практ. зан.	Отчет о выполнении

Из таблицы следует, что ПК-7 полностью реализуется во втором семестре подготовки магистров, а успешность ее формирования у магистрантов напрямую зависит от качества овладения всеми ее компонентами. Так же видно, что в качестве оценочных средств используется отчет о выполненной работе, который может быть оценен экспертом с помощью Таблицы 2.

Таблица 2 – Показатели и критерии оценивания уровня приобретенных компетенций (компонент)

Показатели оценивания	Результаты обучения из РП дисциплины	Критерии оценивания компетенций
Пороговый	Знать	Знает: базовые общие знания
	Уметь	Умеет: основные умения, требуемые для выполнения простых задач
	Владеть	Владеет: работает при прямом наблюдении
Базовый	Знать	Знает: факты, принципы, процессы, общие понятия в пределах области исследования
	Уметь	Умеет: диапазон практических умений, требуемых для решения определенных проблем в области исследования
	Владеть	Владеет: берет ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем
Высокий	Знать	Знает: фактическое и теоретическое знание в пределах области исследования с пониманием границ применимости
	Уметь	Умеет: диапазон практических умений, требуемых для развития творческих решений, абстрагирования проблем
	Владеть	Владеет: контролирует работу, проводит оценку, совершенствует действия работы

Для оценивания других ПК и ОПК может возникнуть необходимость разработки оценочных средств другого вида, как то экзаменационные билеты, контрольные вопросы, тесты, проектные задания для порогового, базового и высокого показателя оценивания. Это очень трудоемкая и сложная работа, которая ложится на преподавателей, реализующих образовательную программу. Но чем лучше будет

проделана эта работа, тем с большей уверенностью мы можем говорить об уровне сформированности компетенций будущих специалистов.

Список литературы

1. Азарова, Р.Н. Разработка паспорта компетенции: Методические рекомендации для организаторов проектных работ и профессорско-преподавательских коллективов вузов. Первая редакция./ Азарова Р.Н., Золотарева Н.М. – М.: Исследовательский центр проблем качества подготовки специалистов, Координационный совет учебно-методических объединений и научно- методических советов высшей школы, 2010. – 52 с.

2. Елисеев, И.Н. Методология оценки уровня сформированности компетенций студентов/ Елисеев И.Н. Информатика и образование, 2012 № 4 (233) ISSN 0234-0453

3. Психологические тесты онлайн [Электронный ресурс]. - Режим доступа: <http://www.psytest.info>



МОДЕЛЬ УПРАВЛЕНИЯ УСТОЙЧИВОСТЬЮ ИНФОРМАЦИОННОЙ СИСТЕМЫ С УЧЁТОМ ЧЕЛОВЕЧЕСКОГО ФАКТОРА

Сидарас А.А.

научный руководитель доц. Краснов И.З.

Сибирский федеральный университет

В современных условиях информатизации общества одним из решающих факторов, определяющих устойчивость функционирования информационной системы, является человеческий фактор, который напрямую зависит от профессионализма пользователей, определяющийся уровнем компетентности каждого относительно требуемого вида деятельности соответствующего бизнес-процесса, основанного на информационных технологиях^[1].

Российский стандарт ГОСТ Р ИСО/МЭК 15408^[5], в котором указаны требования к построению защищённых систем, утверждает, что основное дестабилизирующее воздействие на информационную систему оказывает человек, определяя его как злоумышленника, переводя в категорию инсайдера и определяя его как лицо оказывающее деструктивные воздействия на устойчивость информационной системы.

Международный стандарт СОВИТ 5^[1] рекомендует во главу управления бизнес-процессами, основанными на информационных технологиях ставить людей с высоким уровнем компетентности и обладающих лидерскими качествами. Данное руководство, предлагая учитывать компетентности персонала, не регламентирует то, каким образом, и по какой методике производить расчёт уровня компетентности и не предоставляет методики перевода качественных характеристик в количественные величины.

ГОСТом Р ИСО/МЭК 15408 инсайдер определён как злоумышленник, а в соответствии с СОВИТ5 мы определяем его как человека с низким уровнем компетентности, способным влиять на устойчивость информационной системы.

Для оценки влияния человеческого фактора была взята типология К.Г. Юнга^[6]. В основе типологии лежит психологическая предрасположенность человека к виду деятельности, совместно с образовательными характеристиками она определяет уровень его профессионализма. Типология утверждает, что каждый человек имеет сильные и слабые стороны. Сильные стороны человека лежат в основе предрасположенности человека к конкретному виду деятельности. Наша задача, выявив сильные стороны человека и определив его образовательные характеристики, рассчитать уровень компетентности.

Тестирующая программа <http://www.akmekras.ru>, разработанная студентами выпускных курсов, позволяет выявить сильные стороны человека определяющих его предрасположенность к деятельности^[3]. Сильные стороны рассматриваются как качественные характеристики и определяются как неуправляемая контролируемая переменная. В свою очередь, уровень образования и опыт работы рассматриваются как образовательные характеристики и определяются как управляемая контролируемая переменная. Перевод качественных характеристик в количественные показатели методами сравнительного анализа, с учетом образовательных характеристик (методом анализа иерархий, методом функций полезности, методом непосредственной оценки)^[4], позволяет рассчитать уровень компетентности каждого пользователя относительно требуемой деятельности.

В понятие расчётный уровень компетентности входит предрасположенность к определённому виду деятельности, и ей присваивается цветовой индикатор

предрасположенности к деятельности P_i ($i=1,..,4$) и образовательная характеристика Q_i ($i=1,..,4$), всё относительно вида деятельности D_i ($i=1,..,4$)^[2].

Следует отметить, что относительно P_i и D_i можно брать $i=1..4$; $i=1,..8$; $i=1..16$. $i=1,..,4$ взято для упрощения понимания методики расчёта уровня компетентности.

Соблюдая логику, описанную в таблице 1, лицо принимающее решение (далее ЛПР) относительно устойчивости информационной системы с учётом человеческого фактора может добиться эффективного управляющего воздействия, используя систему поддержки принятия решений (далее СППР), не обладая глубокими знаниями в психологии, математики и информатике.

На систему, согласно рисунку 1, оказывают влияние следующие воздействия:

- $\mu(t)$ – контролируемые управляемые (Q_i) и неуправляемые (P_i) воздействия;
- $u(t)$ – управляющее воздействие на систему.

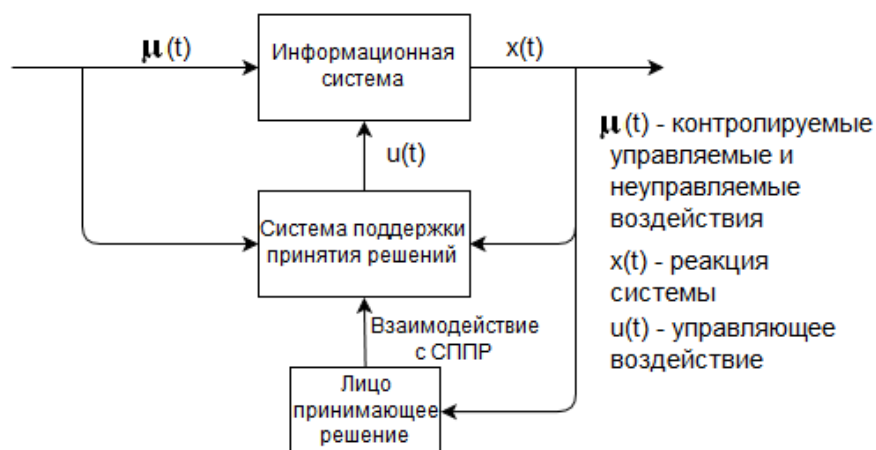


Рис.1 – Схема информационной системы с СППР и ЛПР

Реакция системы (состояние системы) $x(t)$ должна стремиться к идеальной реакции $x^*(t)$. Идеальная реакция (состояние) для информационной системы считается известной и определённой. Логика работы СППР состоит в получении $\mu(t)$ и построении специальной таблицы, которая содержит уровни компетентности и ранжированные ресурсы организации. Данные этой полученной таблицы вычитаются из данных эталонной таблицы состояния. Разности выдаются ЛПР в удобной форме (рис. 2 – снизу). ЛПР выдаёт управление $u(t)$ и через СППР отслеживает принятое решение (состояние системы) $x(t)$.

Таблица 1 – Логика ЛПР

Вид деятельности	Соотношение
D_1	$P_1 > P_4 > P_2 > P_3$
D_2	$P_2 > P_3 > P_1 > P_4$
D_3	$P_3 > P_2 > P_4 > P_1$
D_4	$P_4 > P_1 > P_3 > P_2$

СППР предоставляет ЛПР два состояния информационной системы с учетом расчетного уровня компетентности. Первое состояние показывает то, как должны распределиться уровни доступа в информационной системе в эталонном варианте, обеспечивающее максимальную устойчивость при идеальном соотношении психологических установок человека, образовательных характеристик относительно

требуемого вида деятельности. Второе состояние показывает то, в каком состоянии находится информационная система на текущий момент времени. ЛПР, анализируя два состояния, предоставленные СППР в графическом виде, принимает решение по управленческому воздействию на распределение уровней доступа, на основании матрицы доступа, к ресурсам тем самым обеспечивая устойчивость системы^{[2][3]}. С целью достижения максимальной устойчивости информационной системы ЛПР принимает решение по стабилизации системы, анализируя состояние системы, пошагово распределяя ресурсы на основании расчетного уровня компетентности пользователей, как можно более подходящее к эталонному варианту.

Управляющие воздействия ЛПР выражаются в изменении прав доступа к информационным ресурсам. Например, если $D_1 = P_1$ и Q_1 , то уровень доступа максимальный, если $D_1 = P_4$ и Q_4 , то в данном виде деятельности человек является инсайдером, то есть он приведёт к дестабилизации информационной системы, и, поэтому ЛПР необходимо принять управленческое решение при помощи СППР.

Ниже приведены графики, которые соответствуют эталонному состоянию во всех видах деятельности (слева), реальному состоянию во всех видах деятельности (справа), разностному значению между двумя состояниями системы (снизу):

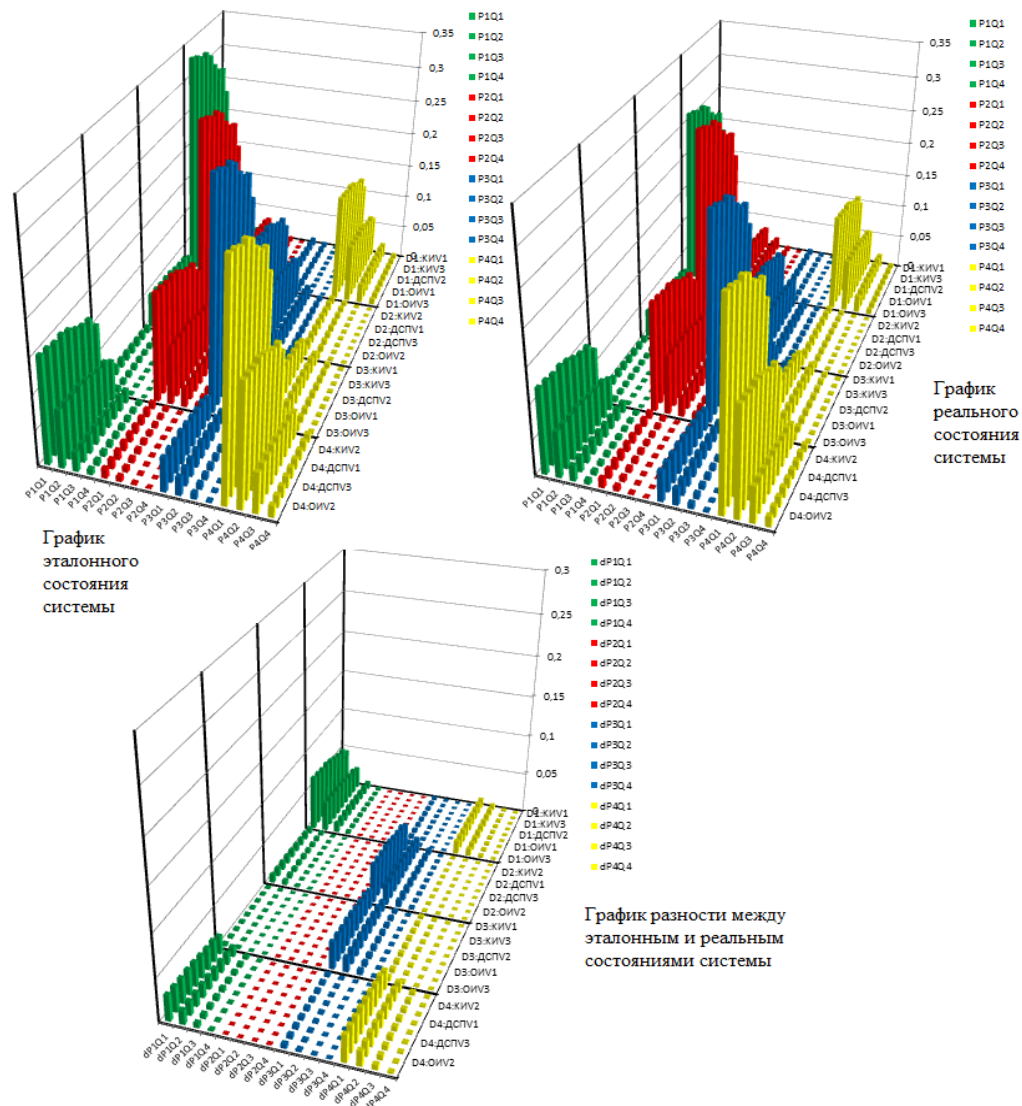


Рис.2 – Графики состояний

Ранжирование расчетных уровней компетентности позволяет выявить уровни лидерства, менеджера и исполнителя в информационной системе присвоив каждому соответствующий ранг и права доступа к информационным ресурсам системы. Результаты, представленные ЛПР в графическом виде, облегчают принятие решения по распределению уровней и прав доступа на основании матрицы доступа с целью управления стабильностью системы^[3].

Список литературы

1. COBIT 5, Бизнес модель по руководству и управлению ИТ на предприятии.
2. И.З. Краснов Модель информационной структуры организации на основе компетентностного подхода // Материалы XIX Международной научно-практической конференции, посвященной 55-летию Сибирского государственного аэрокосмического университета имени академика М. Ф. Решетнева (10–14 ноября 2015, г. Красноярск). Часть 2. С. 227 – 231
3. Краснов И.З. Матрица доступа как пассивный элемент защиты информационных ресурсов. // Материалы XV Всероссийской научно-практической конференции «Проблемы информационной безопасности государства, общества и личности». – Томск – Иркутск. – Томск: В-Спектр . – 2014. – С. 202– 210.
4. Гудков П.А. Методы сравнительного анализа. Учеб. пособие. – Пенза: Изд-во Пенз. гос. ун-та, 2008 – 81 с.; 5 ил., 15 табл., библиогр. 12 назв.
5. ГОСТ Р ИСО/МЭК 15408-2008 «Методы и средства обеспечения безопасности критерии оценки безопасности информационных технологий»
6. Краснов И.З. Дифференцированный подход к подготовке специалистов информационной безопасности. // Доклады VI Пленума СибРОУМО по образованию в области информационной безопасности и XV конференции. – Томск – Иркутск. – Томск: В-Спектр . – 2014. – С. 94– 99.



ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ОБРАБОТКИ И АНАЛИЗА ДЕНДРОХРОНОЛОГИЧЕСКИХ ДАННЫХ

Скрипкин М. А., Дозморов А. И.

научный руководитель канд. пед. наук Виденин С. А.

Сибирский федеральный университет

Ряд научных институтов России и мира в последнее время ведут исследования в области дендрохронологии. В частности, ученые исследуют годовые кольца деревьев. Они обладают внутренней структурой, которая содержит дополнительную экологическую информацию, и она все ещё редко используется. Анализ количественных показателей микроанатомической структуры годовых колец позволит не только повысить точность и разрешающую способность климатических реконструкций, но и оценить функционально-физиологический отклик деревьев на изменчивость климатических условий^[1].

В силу того, что микроанатомическая структура годовых колец попала в область научных интересов исследователей не так давно, связи с информационными науками не были наработаны и, в результате, процесс анализа образцов был слабо автоматизирован. Виртуальную его часть можно описать следующим образом:

— клеточное строение образца считывается с помощью цифрового микроскопа и сохраняется в графический файл, содержащий мета-данные (объемом ~2Гб);

— из полученного файла полуавтоматическим путём (необходимо выбрать область измерения) выделяется информация о ширине и количестве клеток на измеренном участке, которая затем сохраняется в Excel-файл в минимально форматированном виде;

— таблицу с результатами в ручном режиме приводят к удобному формату и продолжают обработку полученных данных (строят графики, высчитывают другие значения и т.д.).

Из вышесказанного легко выделить несколько проблем используемого подхода:

— необходимость хранить сверх-большой (даже по нынешним меркам) файл с исходными данными на случай, если появится необходимость в анализе другого участка образца;

— затруднённый обмен исходными данными между удалёнными исследователями ввиду больших размеров файлов: не все готовы передавать или принимать файлы такого большого объёма; не все сохраняют исходные файлы после получения требуемых на момент исследования данных;

— для открытия как исходных файлов, так и графических файлов, ввиду их большого разрешения, требуется компьютер с большой вычислительной мощностью.

Также стоит отметить, что используемое программное обеспечение разрабатывается для выполнения не связанных с дендрохронологией задач - оно не предоставляет практически никаких возможностей кроме обработки изображений с помощью математических алгоритмов, находящихся в открытом доступе, кроме этого, при каждом обновлении данного программного обеспечения теряется возможность использовать сторонний инструментарий, который был создан на основании предыдущей версии.

В ходе работы научных коллективов Института космических и информационных технологий и Института фундаментальной биологии и биотехнологии было принято



решение разработать и разместить в сети Интернет портал, на котором можно было бы хранить исходные изображения (и дополнительную информацию о них), а также обрабатывать их и, при желании, делиться ими с другими исследователями.

В данный момент существуют сервисы для хранения данных, но они представляют из себя простые FTP-серверы, и не предоставляют средств для обработки загруженной информации. Таким образом, все учёные используют разный инструментарий, получая данные, которые некорректно сравнивать с результатами, полученными другими исследователями. В отличие от существующих систем хранения данных, разрабатываемый сервис предоставит также инструменты для их обработки и анализа, которые будут выдавать пользователю стандартизированную информацию.

В следующей части доклада под термином "публикация" будет пониматься комплекс стандартизированных данных, описывающих определенный участок древесины и являющихся ячейкой структурированного списка подобных им данных, который будет обозначаться термином "галерея".

Общий принцип работы данного сервиса можно описать следующими пунктами:

1. Пользователь регистрируется на сайте и дожидается одобрения его регистрации одним из администраторов.

2. После этого пользователь может загружать и скачивать материалы, касающиеся дендрохронологии.

3. После загрузки изображений клеток, интегрированная система анализа снимков обрабатывает их и формирует файл со всеми измерениями клеток на этом снимке, которые пользователь может впоследствии использовать в своих исследованиях.

4. Пользователь может опубликовать свои снимки и данные, чтобы они отображались в общем списке, а также назначить ограничения на их скачивание.

5. Другие пользователи могут найти опубликованные материалы используя поиск на сайте.

Незарегистрированный пользователь, зайдя на сайт, может лишь просматривать список загруженных материалов с их метаданными, но не может ничего скачивать и загружать. Когда он регистрируется, администратор проверит его регистрационные данные, в которых пользователь должен будет помимо всего указать место своей работы/учебы, и установит статус этого пользователя.

После регистрации появляется возможность пользоваться личным хранилищем и загружать туда материалы своих исследований. Управлять загруженными материалами он сможет, перейдя по ссылке из своего профиля. Там же будет ссылка на список скаченных им материалов.

После проверки администратором, он сможет публиковать свои материалы для всеобщего обозрения, а также скачивать материалы других пользователей, которые они выложили в открытый доступ. Помимо этого, у пользователя появится возможность использовать функцию анализа загруженных им снимков.

Рассмотрим более подробно механизм публикации данных на сервисе.

На странице галереи, помимо самого списка данных, имеется несколько кнопок: две для сортировки галереи по дате добавления данных и популярности, а также одна для размещения новой публикации.

Решив создать новую публикацию, пользователь должен будет выбрать из личного хранилища интересующий его снимок клеток, файл с измерениями, получить который можно воспользовавшись функцией анализа снимков, а также другие файлы, которые он считает нужным прикрепить к этой публикации. Помимо файлов, пользователь должен



указать обязательные параметры загружаемых данных, такие как вид дерева, континент, страна и координаты его расположения, природную зону, год сбора, а также публикации, со ссылками на них, в которых использовались данные материалы. На основании снимка будет создано превью-изображение, которое будет доступно для всеобщего обозрения.

После этого автор может выставить ограничения этой публикации для других пользователей. Есть три возможных варианта: публикация без ограничений, то есть любой пользователь, прошедший модерацию может его скачать; публикация полностью ограниченная; публикация с частичным ограничением – пользователь, которого заинтересовали данные материалы, оставляет заявку, которая информирует по электронной почте автора публикации, что к его измерениям хотят получить доступ, автор может связаться с заинтересованным лицом или сразу подтвердить запрос.

Чтобы найти в галерее что-то конкретное, необходимо воспользоваться поиском на сайте. В нем требуется указать интересующие параметры, которыми являются метаданные (вид дерева, год сбора и прочие), и система выдаст все совпадения.

В файлах с измерениями, полученными в результате анализа снимков, содержится вся возможная информация о клетках: ширина их стенок, ширина люмена, площади и т.п. Также, имеется возможность конкретизировать запрос и, вместо файла со всей возможной информацией, получить, к примеру, только площади люменов с 1950 по 1970 год.

В процессе анализа накопленных данных часто возникает необходимость построения зависимостей между различными параметрами, число которых может быть значительным. Возможность анализа зависимостей между различными параметрами предполагает возможность представления данных в виде многомерной модели – гиперкуба, или OLAP-куба, в ячейках которого хранятся анализируемые данные, например толщина люмена клетки. Оси куба представляют собой измерения, по которым откладываются параметры, относящиеся к анализируемой предметной области, например названия месяцев года. В простейшем случае двумерного куба получается таблица, показывающая значения толщины люмена клетки по месяцам. Дальнейшее усложнение модели данных возможно по нескольким направлениям:

1. Увеличение числа измерений - данные о толщине люмена клетки не только по месяцам, но и по регионам. В этом случае куб становится трехмерным.

2. Усложнение содержимого ячейки - например, нас может интересовать не только толщина люмена, но и толщина стенки. В этом случае в ячейке будет несколько значений.

3. Введение иерархии в пределах одного измерения - общее понятие «время» связано с иерархией значений: год состоит из кварталов, квартал из месяцев и т.д.

В дальнейшем эти данные используются для осуществления интеллектуального анализа данных или data mining.

В заключении, в таблице 1 представлена сравнительная характеристика решений – существующих и разрабатываемого сервиса.



Таблица 1 – Сравнительная характеристика решений

Критерий сравнения	Существующие решение	Разрабатываемый сервис
Наличие стандартов	Нет единой стандартизации, что затрудняет ученым использовать эти данные.	Стандартизированный файл с измерениями благодаря автоматическому анализу снимков клеток.
Объем данных	Объёмы, доступные для хранения данных, зависят от возможностей каждого отдельного пользователя	Возможно сохранение большого количества снимков объёмом ~2Гб.
Поиск данных	Поиск по тексту. Результат не гарантирован, т.к. наличие каких-либо комментариев и пояснений при сохранении данных не требуется. Минимальное количество атрибутов, к примеру вид дерева.	Гарантированный успех поиска по метаданным при наличии самих данных благодаря требованию ввода обязательных метаданных перед публикацией. Предварительный просмотр превью-изображения снимка.
Анализ снимков	В основном платные решения, которые как и бесплатные слабо автоматизированы и не могут дать нужного результата. Необходимость иметь высокую вычислительную мощность у машины, на которой будет проводиться анализ. Результаты анализа требуют последующей обработки перед использованием.	Анализ снимков будет проходить автоматически и будет соответствовать всем требованиям. Результат не требует последующей обработки. Анализ проводится на сервере, поэтому от клиента требуется только предоставить снимок. Высокая скорость анализа.

Проанализировав эту таблицу можно сделать вывод, что разрабатываемый сервис гораздо эффективнее существующих решений. Он возлагает минимальные требования на пользователя, автоматически проводит анализ снимков за короткое время, отвечает необходимой точности анализа и отображения его результатов, отличается стандартизованностью.

Список литературы

1. Dendro-Chronological Dating of Constructions in the Course of Archeological Excavations of the Old-Turukhansk Site (New Mangazeya) // Journal of Siberian Federal University. Humanities & Social Sciences. 2011. Vol. 4. No 7. P. 952–963
2. Демченко А.А., Молоков В.В. OLAP-технология анализа данных. Журнал Актуальные проблемы авиации и космонавтики, Выпуск № 10 / том 1 / 2014.



**ФОРМИРОВАНИЕ МУЛЬТИЛИНГВИСТИЧЕСКОГО
ИНФОРМАЦИОННО-ТЕРМИНОЛОГИЧЕСКОГО БАЗИСА ДЛЯ
АДАПТИВНЫХ ОБУЧАЮЩИХ СИСТЕМ**

Старовойтов А.М

научный руководитель канд. техн. наук Корнеева А.А.

Сибирский федеральный университет

Современного квалифицированного специалиста трудно представить без знания иностранного языка, без умения ориентироваться в зарубежной литературе по той отрасли, в которой он работает. В связи с этим возникает вопрос об эффективности обучения языку.

В рамках решения этой проблемы обучение иностранному языку во многих неязыковых ВУЗах имеет профессиональную ориентацию. Одним из основных моментов в таком обучении является освоение характерной для учебной специальности иностранной терминологии.

Однако требование повысить уровень обучения иностранным языкам в неязыковом вузе вступает в противоречие с некоторыми условиями этого обучения, одним из которых является ограниченное количество часов, отводимых на изучение языка. Выходом из этого противоречия может стать лишь оптимизация процесса обучения. Для этого необходимо не только найти эффективные, наиболее экономные приемы обучения языку, но и определить тот лингвистический материал, на котором можно успешно применить эти приемы.

Методика должна строиться с учетом обучения не языку вообще, в целом, а с учетом выработки у обучаемых определенных навыков: например, в разговорно-бытовой речи, в письме, в понимании (переводном или беспереводном) текста – художественного, научного, публицистического и т.д. Усилия преподавателя и обучаемого должны фокусироваться на достижении конкретной цели, на умении использовать знание иностранного языка в определенных ситуациях. Человека, работающего в технической области, прежде всего, интересует возможность понимать текст по своей специальности, а умение вести беседу о музыке или кино будет иметь для него второстепенное значение. Компьютеризованные учебные материалы (учебные компьютерные программы) способны полнее и глубже адаптироваться к индивидуальным особенностям студентов.

Обучения языку, с помощью обучающих систем позволяет решить проблему эффективности обучения иностранным языкам, позволяет оптимизировать процесс обучения, а также позволяет максимально активизировать самостоятельную работу студентов, что в условиях вузовского обучения особенно важно.

Развитие информационных технологий на сегодняшний день позволяет разработать эффективный программно-алгоритмический аппарат для создания компьютерных адаптивных обучающих систем, в том числе, для изучения иностранной терминологической лексики.

На данный момент существует несколько мультилингвистических адаптивных систем обучения иностранному языку, основанных на формировании частотных словарей[2], а также на основе словарей, сформированных по принципу схожести слов(Паронимов). Планируется разработка с улучшением основных показателей работы системы, а именно показателей интерактивности, унификации, производительности и стабильности.

На основании требований к уже существующим мультилингвистическим системам обучения, а также в ходе исследований были сформулированы следующие требования к разрабатываемой системе:

1. Реализация такой связи учебного материала (компьютерной программы) с обучаемым, которая ведется в диалоговом режиме, имитируя некоторые функции преподавателя и, в известной степени, общение.

2. Наличие обратной связи, под которой мы понимаем возможность осуществления коррекции ответа самим обучаемым с опорой на консультирующую информацию. Консультирующая информация выбирается из памяти компьютера либо самим студентом, либо на основе автоматической диагностики ошибок, допускаемых студентом в ходе работы. Способ предъявления подобного рода информации зависит от типа учебной компьютерной программы.

3. Из возможности диагностики ошибок вытекает и такая особенность компьютера (точнее, компьютерной программы), как адаптивность. Компьютеризованный урок проходит с учетом индивидуальных особенностей студентов. Изучение, тренировка, повторение и контроль одного и того же материала может осуществляться:

- с различной степенью глубины и полноты;

- в индивидуальном темпе;

- в индивидуальной последовательности (выбираемой самим студентом).

4. Возможность проводить в автоматическом режиме многофакторный сбор и анализ статистической информации, получаемой в процессе компьютеризованного занятия, без нарушения естественности протекания компьютерного урока. При этом компьютер способен фиксировать достаточно большое количество параметров:

- время, затраченное студентами на работу со всей программой, группой заданий или с каким-либо конкретным заданием или упражнением;

- количество верных/неверных ответов (причем, неверные ответы могут «раскладываться по полочкам» в памяти компьютера);

- количество обращений к справочной информации, а также характер наиболее часто запрашиваемой помощи той или иной группой обучаемых;

- количество попыток при выполнении заданий. Компьютер может фиксировать информацию и по другим параметрам. Это определяется авторами учебных программ в зависимости от целей и задач, выдвигаемых педагогами.

На сегодняшний день ведется работа по разработке и исследованию алгоритма разбиения базиса слов на модули оптимальные по трудоемкости.

Из всего вышесказанного следует что проблема эффективности обучения иностранному языку актуальна. Достигнутые на сегодняшний день результаты позволяют нам говорить о том, что система будет эффективна, надежна и позволит расширить круг подходов для решения проблемы эффективности обучения иностранному языку.

Список литературы

1 Голубева Т.И., Репина С.О. Применение информационных технологий в обучении иностранному языку: Учебное пособие. - Оренбург: ГОУ ОГУ, 2004. - с. 9-13.

2 Огнерубов С.С., Лесков В.О. Мультилингвистическая система обработки текстов на базе скрытой Марковской модели // Вестник НИИ СУВПТ, 2006, выпуск 24.- С.9-19



АНАЛИЗ ГИДРОЛОГИЧЕСКОГО РЕЖИМА РЕК СИБИРСКОГО РЕГИОНА С ИСПОЛЬЗОВАНИЕМ ГИС

Торгашин Р.К.

научный руководитель канд. физ.-мат. наук Картушинский А.В.

Сибирский федеральный университет

На территории Сибирского региона находится большое количество рек, которые играют огромную роль в большинстве отраслей экономики региона. И поэтому необходимость анализа и контроля изменчивости гидрологического режима водных объектов становится все более актуальной задачей в последнее время. Решение данной проблемы позволит снизить вероятность возникновения чрезвычайных ситуаций связанных с осенне-весенними половодьями, позволит прогнозировать появление опасных гидрологических процессов, способных причинить ущерб экономике региона, и разработать мероприятия по предотвращению неблагоприятных последствий и минимизации ущерба на водных объектах.

Анализ изменчивости гидрологического режима рек Сибирского региона должен опираться на информационные и программные средства, включающие в себя базу данных наблюдений и программное обеспечение для применения методов статистического анализа.

Наиболее перспективным направлением информационного обеспечения научно-практической деятельности является разработка тематических геоинформационных систем (ГИС), в основе которых лежит электронное (цифровое) картографирование. Геоинформационные системы с каждым годом применяются во многих отраслях, в том числе и в области инженерной гидрологии.

Происходящие и ожидаемые изменения климата в первой половине XXI века являются причиной изменения водохозяйственной обстановки. В ряде регионов России эти изменения носят как положительный, так и отрицательный характер. Одним из отрицательных явлений является повышение риска возникновения опасных гидрологических процессов (особенно наводнений), обусловленных изменением составляющих водного баланса речных бассейнов. Наводнения являются одним из наиболее часто повторяющихся стихийных бедствий, а по площади охватываемых территорий и наносимому ущербу превосходят все другие чрезвычайные ситуации. Для всестороннего анализа развития и последствий затопления территорий необходимо привлечение большого объема пространственной информации: о регионе в целом, затопляемых территориях и об объектах, подверженных затоплению.

Кроме того, учитывая развитие нефтегазовой отрасли, необходимы оценки возможного воздействия гидрологического режима рек на нефте- и газопроводов в местах их расположения.

Для обработки такого рода информации необходимо использовать геоинформационные системы (ГИС), так как они объединяют в себе возможности автоматизированных картографических редакторов и систем управления базами данных.

В данной работе анализ выполняется в программной среде ArcGIS. Этот геоинформационный программный продукт представляет собой инструмент для построения ГИС любого уровня. ArcGIS дает возможность легко создавать данные, карты, глобусы и цифровые модели в настольных программных продуктах, затем публиковать их и использовать в настольных приложениях, в веб-браузерах, и через мобильные устройства. У этой программы имеется ряд преимуществ, благодаря

которым можно анализировать и обрабатывать различные данные с применением следующих процедур:

1. Анализ пространственно распределенной информации;
2. Управление данными;
3. Картографирование и визуализация объектов;
4. Расширенное редактирование свойств объектов;
5. Послойный обмен данными;
6. Кастомизация - для разработки пользовательского интерфейса с наиболее удобными настройками и оформлением в соответствии с поставленными задачами и предпочтениями.

Говоря о системном анализе гидрологического режима рек Сибирского региона, необходимо понимать какие факторы оказывают влияние на динамику гидрологических показателей рек. Гидрологический режим реки представляет собой закономерные изменения состояния водного объекта во времени (уровня и расхода воды, ледовых явлений и т. д.), обусловленные физико-географическими свойствами бассейна и, в первую очередь, климатическими особенностями данного бассейна [1]. На изменения гидрологического режима реки основное влияние оказывает изменение гидрологических показателей водоема.

К основным гидрологическим показателям относятся:

1. Годовой сток воды, л/сек км²;
2. Максимальный сток воды весеннего половодья и дождевых паводков, л/сек км²;
3. Минимальный сток воды рек, л/сек км²;
4. Наивысший уровень воды в реке, м;
5. Характерные уровни воды, высшие для рек с устойчивым ледоставом, м.

Уровень воды – один из самых важных гидрологических показателей, который позволяет специалистам сразу оценить режим реки в данный момент, сказать, опасен ли подъем воды в реке для населения, для объектов, расположенных в пойме. Уровни измеряют постоянно в фиксированных местах – на водомерных постах. Эти посты расположены таким образом, чтобы проводить измерения периодически и с наименьшими погрешностями [1].

В формировании гидрологического режима рек исследуемого региона участвует сложный комплекс физико-географических условий. К числу основных факторов, определяющих особенности структуры водного баланса и гидрологического режима территории, относятся климат, рельеф и почвенно-растительный покров. Эти факторы тесно связаны между собой и воздействуют не только непосредственно, но и через другие факторы. Изучение закономерностей формирования водных ресурсов территории требует учета всего комплекса локальных географических и метеорологических условий. Наибольшее влияние на динамику уровня воды в реке оказывают различные метеорологические параметры [2, 3, 4].

В данной работе реализуется технология обработки гидрологических данных и метеорологических параметров с использованием геоинформационной системы. В качестве основных метеорологических показателей, оказывающих влияние на изменчивость гидрологического режима рек Сибирского региона, выбраны среднесуточные значения температуры воздуха и количества выпавших осадков за соответствующие периоды наблюдения. Структурная блок-схема геоинформационной системы «ГидроГИС» показана на рисунке 1.



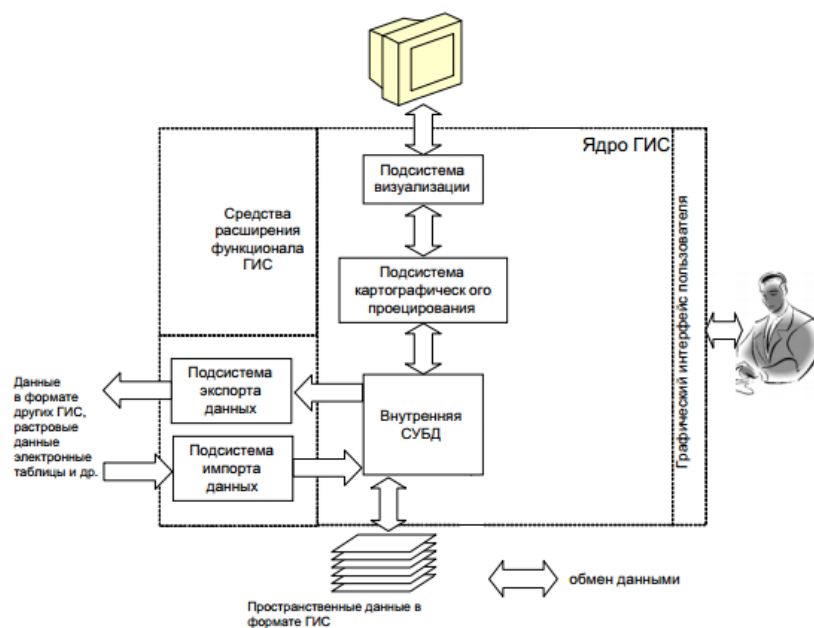


Рис.1 - Структурная блок-схема геоинформационной системы «ГидроГИС»

Перспективы развития такого рода ГИС связаны с тем, что на территории Сибирского региона находится большое количество рек, которые играют огромную роль в большинстве отраслей экономики региона [1], в том числе и нефтегазовой отрасли. Разработка геоинформационных систем предназначенных для анализа и контроля изменчивости гидрологического режима водных объектов становится все более актуальной задачей в последнее время. Решение данной проблемы позволит:

- провести оценку вероятности возникновения чрезвычайных ситуаций связанных с осенне-весенними половодьями;
- создаст возможности для прогнозирования появления опасных гидрологических процессов, способных причинить ущерб экономике региона;
- разработать мероприятия по предотвращению неблагоприятных последствий и минимизации экологического и экономического ущерба.

Список литературы

1. Лысенко, В.В. Гидрологические исследования в Сибири [Текст] / под ред. В. В. Лысенко, В.П.Ергина. - М.: Гидрометеиздат, 1991. - 115 с.
2. Пьянков, С.В. Мониторинг водохозяйственных объектов с использованием ГИС-технологий (на примере пруда с. Ножовка Чагинского района Пермской области) [Электронный ресурс] / Пьянков С.В., Шавнина Ю.Н.// ГИС центр ПГУ. – 2007. – Режим доступа: <http://www.gis.psu.ru>.
3. Карты областей России, городов, туристические карты, локации рек, схемы пещер [Электронный ресурс]. – 2008. – Режим доступа: <http://www.pohodmaps.ru>.
4. Климатическая классификация рек [Электронный ресурс]. – 2006. – Режим доступа: <http://www.detskiysad.ru/raznlit/zemlevedenie086.html>.
5. Матюшкин Д.Ю., Картушинский А.В. Анализ изменчивости гидрологического режима рек Сибирского региона с использованием ГИС [Текст] // Молодежь и наука: начало XXI века: сб. материалов V Всероссийской научно-технической конференции студентов, аспирантов и молодых ученых: в 7 ч. Ч. 2. МИОЦ ФГОУ ВПО СФУ. – Красноярск, 2009. - С. 144 – 146.

ОПТИМИЗАЦИЯ СЕТИ ПЕРЕДАЧИ ДАННЫХ НА ПРИМЕРЕ СИБИРСКОГО БАНКА ПАО СБЕРБАНК

Тюрюмина М.Д.

научный руководитель канд. техн. наук Даничев А.А.

Сибирский федеральный университет

С каждым годом объемы передаваемой информации посредством сети Интернет стремительно растут. Соответственно возрастает нагрузка на сети передачи данных, что служит толчком для их усовершенствования и оптимизации.

Интернет-провайдеры (далее провайдеры) стремятся быть конкурентоспособными, поэтому качество предоставляемых услуг улучшается, изменяется стоимость аренды каналов, появляются новые каналы связи и т.д.

Иногда изменения в тарифных планах существенны и выгодны для пользователей, но проанализировать их и решить, стоит ли воспользоваться теми или иными услугами довольно проблематично.

Рассмотрим Сибирский банк. Он является филиалом ПАО Сбербанк и работает на территории Красноярского края, Новосибирской, Томской, Кемеровской области, Республики Алтай, Тыва и Хакасии [3].

Для улучшения качества обслуживания клиентов банк активно развивает альтернативные каналы продаж и постоянно работает над расширением спектра услуг на устройствах самообслуживания и банкоматах. Список получателей платежей постоянно расширяется. Услуга приема платежей населения реализована как на банкоматах, так и информационно-платежных терминалах Банка. Связь между банкоматами, терминалами, отделениями и центром обработки данных обеспечивается с помощью арендованных у провайдера каналов.

Проанализировав деятельность отдела связи и телекоммуникаций было выявлено, что существует нехватка программного обеспечения, готового анализировать предложения провайдеров и предоставлять альтернативные решения при определении плана аренды дополнительных каналов для улучшения технических характеристик сети передачи данных. Так же была выявлена потребность в настройке специфических для данного банка параметров, влияющих на принятие решения, с целью получения более точного результата.

Банк заинтересован в получении качественных услуг за минимально возможную стоимость. В этом случае необходимо учитывать минимум два различных критерия. Таким образом, можно определить задачу бикритериальной оптимизации.

Топология сети передачи данных представлена в виде графа $G = (V, E)$, где множество вершин $V = \{v_1, \dots, v_n\}$ представляет собой узлы коммутации сети, а множество ребер $E = \{e_1, \dots, e_m\}$ соответствует каналам передачи данных [1].

Оптимизация сети передачи данных может происходить как за счет структуры, так и за счет смены тарифа или провайдера.

Для получения оптимальной структуры сети был использован метод ветвей и границ, который превосходит по времени работы алгоритм полного перебора [2]. Он является достаточным для решения задачи на уровне территориального банка.

Сибирский банк арендует выделенные каналы связи у провайдеров. Только в г.Красноярске на март 2016 года насчитывается более сорока провайдеров [4]. Каждый из них предлагает несколько тарифных планов. Проанализировать такой большой объем данных без ЭВМ и специального ПО крайне затруднительно. Одним из решений данной задачи является создание унифицированной базы данных и реализации

функционала, способного анализировать и определять наиболее выгодные предложения на конкретный момент времени с учетом минимальной требуемой пропускной способности и выделенного бюджета на заключение договоров об аренде выделенных каналов связи.

Список литературы

1. Вишневецкий В.М. Теоретические основы проектирования компьютерных сетей. Техносфера, 2003. С.512.

2. Лазарев Е. А. Метод ветвей и границ для оптимизации структуры сети передачи данных. Известия волгоградского государственного технического университета, № 14. С.189 – 193.

3. Региональная сеть ПАО Сбербанк [электронный ресурс]. Режим доступа: <http://www.sberbank.ru/ru/about/today/territory/>

4. Telekomza - информационный проект о связи и телекоммуникациях в России. [электронный ресурс]. Режим доступа: http://prov.telekomza.ru/providers/sibir_fo/



К ЗАДАЧЕ РАСПОЗНАВАНИЯ ОБРАЗОВ ПО ВЫБОРКЕ НАБЛЮДЕНИЙ БЕЗ УЧИТЕЛЯ

Хохлов А.Н., Сотникова А.П.

научный руководитель канд. техн. наук Корнеева А.А.

Сибирский федеральный университет

В современном мире на первый план выходят наука, компьютерные технологии, информационные системы, управление и прочие направления, важные для развития новых технологий. Именно поэтому на сегодняшний день большую роль играет обработка данных, используемая во многих сферах деятельности. Одним из методов обработки данных является кластерный анализ. Спектр применений кластерного анализа очень широк: его используют в археологии, медицине, психологии, химии, биологии и других дисциплинах. Кластеризация^[1] – это автоматическое разбиение элементов некоторого множества на заранее неизвестное количество групп (кластеров) по принципу схожести. Разбиение выборки на группы схожих объектов позволяет упростить дальнейшую обработку данных и принятие решений, применяя к каждому кластеру свой метод анализа. Пример кластеризации можно увидеть на рисунке 1.

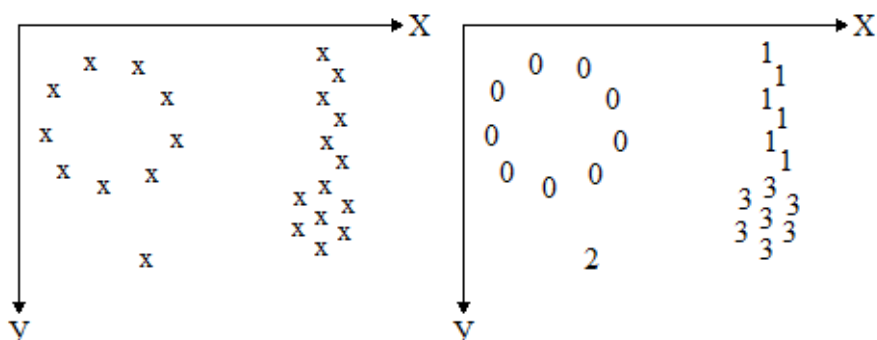


Рис.1 – Пример кластеризации

На сегодняшний день существует большое количество различных алгоритмов кластеризации. В своей работе для решения задачи кластеризации мы использовали алгоритм FOREL и кластеризатор на основе самоорганизующейся нейронной сети Кохонена. В качестве кластеризуемой выборки был взят массив данных под названием Ирисы Фишера (от англ. – Fisher's Iris data set), взятые из репозитория UCI^[2]. Это одна из самых известных баз данных, используемых для распознаваний образов. Набор данных состоит из 150 экземпляров цветков ириса, по 50 экземпляров каждого из трех видов:

- Ирис Щетинистый (от англ. Iris Setosa);
- Ирис Виргинский (от англ. Iris Virginica);
- Ирис Разноцветный (от англ. Iris Versicolor).

Для каждого экземпляра измерялись четыре характеристики (в сантиметрах):

- длина наружной доли околоцветника (от англ. sepal length);
- ширина наружной доли околоцветника (от англ. sepal width);
- длина внутренней доли околоцветника (от англ. petal length);
- ширина внутренней доли околоцветника (от англ. petal width).

Отображение многомерных данных на плоскости является затруднительным, поэтому на рисунке 2 изображено распределение объектов выборки относительно двух

параметров: ширины и длины внутренней доли околоцветника. Данная проекция позволяет в достаточной степени увидеть плотность распределения объектов и их расположение относительно друг друга.

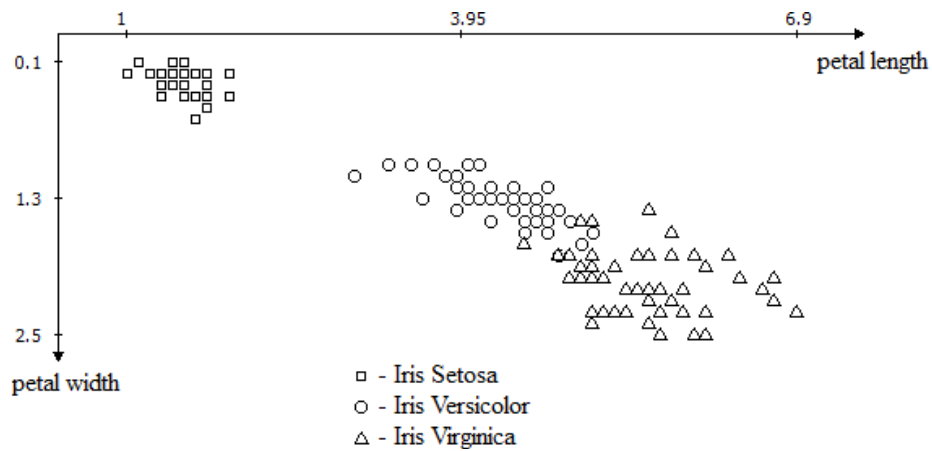


Рис.2 – Распределение данных об ирисах в срезе

В описании к базе данных сказано, что ирисы Iris Setosa линейно отделимы от остальных классов ирисов. Это можно увидеть как на рисунке выше по признаку длины или ширины лепестка, так и по любому другому набору из двух признаков. В тоже время ирисы оставшихся классов Iris Versicolor и Iris Virginica неотделимы линейно один от другого ни по одному из шести возможных наборов из двух признаков.

Для первоначального выделения кластеров воспользуемся довольно простым алгоритмом FOREL. Алгоритм FOREL^[3] основан на идее объединения в один кластер объектов в областях их наибольшего сгущения. На каждом шаге алгоритма строится гиперсфера, которая движется в сторону локального сгущения элементов выборки и постепенно уменьшается в размере, пока состав внутренних точек не останется неизменным. Уменьшение радиуса сфера зависит от параметра α : меньшее значение позволит получить большее количество кластеров. Результат работы алгоритма представлен на рисунке 3.

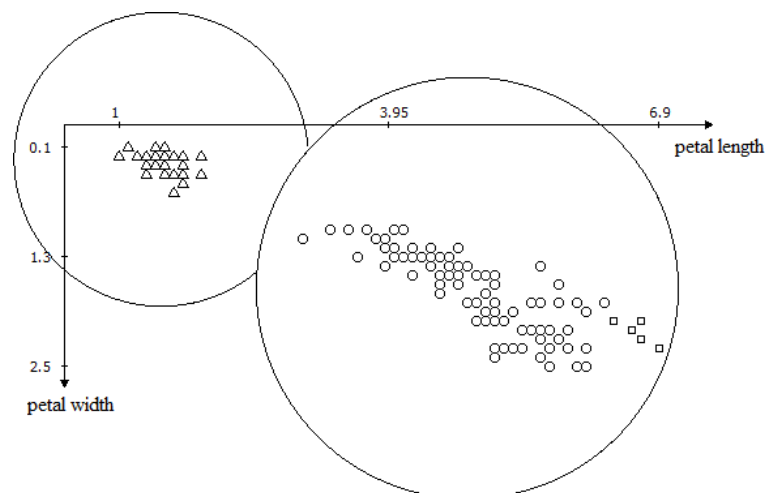


Рис.3 – Результат кластеризации

Варьируя параметр α , одним из лучших вариантов решения является разделение выборки на два кластера, при этом совпадение результатов с ожидаемым ответом доходит до 60%. Это связано с тем, что алгоритм плохо применим к выделению нескольких близко расположенных групп объектов.

В отличие от алгоритма FOREL, использование нейронной сети^[4] позволяет выделить кластеры произвольной структуры и определенного количества. Она состоит из нейронов входного и выходного слоя. Нейроны второго слоя соревнуются друг с другом за право стать наилучшим образом для входного вектора сигнала. Таким образом, при обучении сети каждый из нейронов сдвигается в сторону скопления элементов выборки и образует отдельный кластер, а каждый объект выборки относится к ближайшему для него нейрону.

На рисунке 4 представлен результат работы сети, состоящей из 25 нейронов. Места, отделенные линиями, представляют собой область, которую охватывает каждый из нейронов. Часть из них находится между предполагаемыми кластерами и не содержит ни одного элемента выборки. Зоны, отмеченные разными цветами, содержат в себе как минимум один элемент. Каждая зона одного цвета объединяется в единый кластер, используя поиск в ширину.

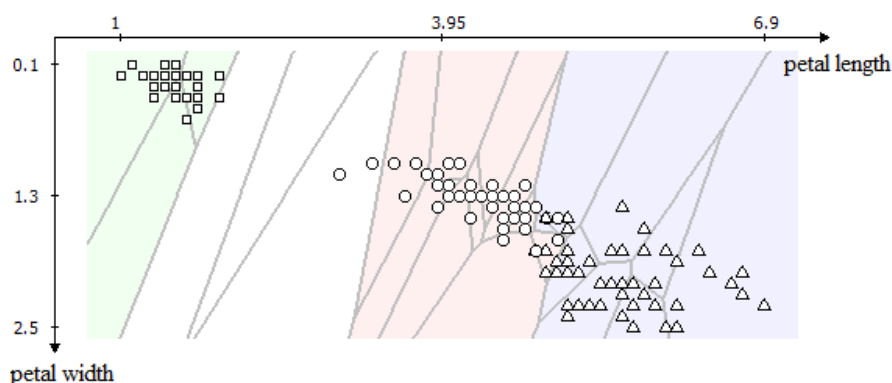


Рис.4 – Результат кластеризации

Совпадение результатов эксперимента с ожидаемыми результатами составляет примерно от 80% до 90% на разных итерациях обучения ввиду того, что такая конфигурация сети с использованием большого количества нейронов позволяет увеличить детализацию результатов процесса кластеризации.

Таким образом, рассмотренные алгоритмы, может, и не дают верного решения поставленных задач, но позволяют выдвинуть предположения о наличии кластерной структуры или числе кластеров, зависимостях между значениями отдельных переменных. В дальнейшем планируется продолжить работу в данном направлении, рассмотреть другие методы с модификациями и достичь лучших результатов.

Список литературы

1. Айвазян С.А., Бежаева З.И., Староверов О.В. Классификация многомерных наблюдений. - М.: Статистика, 1974. - 240с. - С. 75-76.
2. UCI Machine Learning Repository[электронный ресурс]. Режим доступа: <https://archive.ics.uci.edu/ml/datasets/Iris>
3. Загоруйко Н. Г. Прикладные методы анализа данных и знаний. - Новосибирск: ИМ СО РАН, 1999. - С. 38.
4. Нейронная сеть Кохонена[электронный ресурс]. Режим доступа: http://www.machinelearning.ru/wiki/index.php?title=Самоорганизующаяся_карта_Кохонена

СРАВНЕНИЕ МУЛЬТИВЕРСИОННОГО ПРОГРАММИРОВАНИЯ И БЛОКОВ ВОССТАНОВЛЕНИЯ

Черниговский А.С.

научный руководитель канд. техн. наук Царев Р. Ю.

Сибирский федеральный университет

Принципиально, оба подхода, и мультиверсионное программирование, и блоки восстановления обеспечивают основу для достижения отказоустойчивости программного обеспечения. На практике, проектировщик программного обеспечения должен принять решение, относительно того, какой метод будет более эффективен для конкретной программной системы. Чтобы раскрыть относительные преимущества и недостатки этих двух схем ниже проведено сравнение с точки зрения их способностей к обнаружению ошибок, склонностей поощрять эффективные разнообразные способы разработки, а также издержек времени выполнения, которые вероятнее всего возникают из-за использования данных подходов.

Обнаружение ошибок

Практическая эффективность любой отказоустойчивой схемы программного обеспечения критически опирается на ее возможность обнаружить ошибки во внутреннем состоянии системы, прежде чем они смогут повлиять на окружающую среду системы. Проверка голосованием в мультиверсионном программировании обеспечивает мощный подход обнаружения ошибок, где могут быть определены точные результаты. Однако, в случае, если есть потребность в неточном голосовании, возникает сложность проверки голосованием, что может снизить эффективность данного подхода. Тем не менее, приемочные испытания для блоков восстановления зачастую гораздо сложнее разработать и, во многих случаях, они не дают никакой гарантии того, что модуль выполняется правильно. В какой-то степени ошибочность приемочных испытаний может быть уменьшена за счет выполнения утверждений времени исполнения и проверок времени выполнения виртуальной машины. Кроме этого, использование вложенных блоков восстановления не означает, что ошибка, которая не обнаруживается на низком уровне, может быть идентифицирована с помощью приемочных испытаний более высокого уровня. Тем не менее, движение в сторону приемочных испытаний, которые направлены на установление корректности времени выполнения программы, является весьма желательным, и использование формальных методов для выявления таких тестов, вероятно, имеют здесь значение [Melliar Smith 1983]. Только тогда, когда приёмное тестирование основано на таких понятиях корректности, его можно считать соперником эффективной проверки голосованием.

Разнообразие программного обеспечения

Для того чтобы обеспечить бесперебойное обслуживание при наличии остаточных дефектов проектирования программного обеспечения, и мультиверсионное программирование, и блоки восстановления требуют разнообразия программных модулей. Обе схемы могут достичь такого разнообразия, пытаясь произвести независимые программы по одной спецификации, но блоки восстановления предлагают дополнительные подходы, такие как приоритетные полностью функциональные конструкции, где разнообразие сознательно планируется за счет эффективности проектирования и слабые конструкции, где разнообразие достигается за счет функциональности. В частности, вырожденный случай нулевых альтернатив в блоках восстановления может обеспечить эффективную стратегию отказоустойчивости для

систем реального времени, в то время как базовый мультиверсионный подход не имеет аналогов. Тем не менее, там, где должны присутствовать независимые модули, блоки восстановления страдают от ограничения, что алгоритмы для каждого модуля должны работать с одними глобальными структурами данных, и это ограничивает степень независимости, с которой такие конструкции могут быть воссозданы. Способность модулей мультиверсионной схемы локально сохранять свои собственные структуры данных позволяет достичь более высокого уровня независимости конструкций, но некоторое нарушение независимости возникнет из-за спецификации значений перекрестной проверки для проверки голосованием.

Издержки времени выполнения

Высокая надежность программного обеспечения обычно является основным требованием встраиваемых компьютерных систем, в которых необходимость поддерживать отклик в режиме реального времени критична. В таких системах издержки времени выполнения, вызванные мультиверсионным программированием или блоками восстановления, будут являться важным фактором для проектировщика программного обеспечения. Если целевое аппаратное обеспечение для системы содержит один процессор, мультиверсионное программирование будет налагать высокие издержки времени выполнения, так как все отдельные версии должны выполняться последовательно. Однако, когда отдельные аппаратные процессоры доступны для выполнения каждой версии одновременно, производительность мультиверсионной схемы будет соответствовать времени выполнения самой медленной версии, принимающей участие в проверке голосованием (синхронизационный штраф) и продолжительности самого голосования.

В противоположность этому, время выполнения блока восстановления составляет, как правило, работу основного модуля, выполнение приемочных испытаний, а также операции, необходимые для установления и сброса точки восстановления. Однако, при обнаружении ошибки, обратное восстановление, серийное альтернативное исполнение и повторное приёмочное тестирование приведет к увеличению издержек во время выполнения.

Таким образом, издержки блоков восстановления менее предсказуемы, чем издержки мультиверсионных программ, и это должно быть учтено в критичных по времени приложениях, гарантируя достаточное количество запасного времени цикла обработки системы для того, чтобы справиться с наихудшим случаем накладных расходов.

В действительности специальный вариант блока восстановления, известного как механизм крайнего срока [1], специально был предложен для решения этой проблемы. Тем не менее, даже в рамках ограничений общей схемы блока восстановления, многое можно сделать, чтобы уменьшить как масштабы, так и изменчивость издержек времени выполнения. Например, аппаратная поддержка восстановления основной памяти позволит сократить накладные расходы на операции по восстановлению; кэш восстановления [2] был успешно разработан для этой цели, также был разработан ещё один метод, основанный на применении внутренне восстанавливаемых устройств памяти, который предложил [3]. Более того, функционально размытые альтернативы приведут к сокращению издержек выполнения, а также использование нулевых альтернатив может быть эффективным вариантом для систем реального времени. Тем не менее, более острая и основная издержка времени выполнения блоков восстановления – это синхронизационный штраф структур коммуникационного типа, используемых для координации восстановления между параллельными процессами. Здесь упор делается на проектировщиках программного обеспечения, которые принимают решение о выборе структур коммуникации, которые, в свою очередь,



сказываются на естественной синхронизации конструкций, тем самым сводя издержки к минимуму [4].

Заключение

Из этих сравнений видно, что выбор схемы должен зависеть от характеристик самой программной системы: необходимость реагирования в реальном времени, требования к наличию точек проверки корректности вычислений, возможность вычисления точных показателей надёжности системы и др., так и конфигурации аппаратного обеспечения, на котором она будет работать. В целом, критическое значение обнаружения ошибок в отказоустойчивом программном обеспечении предполагает, что в том случае, когда можно легко построить проверки голосованием, а также имеется необходимое количество аппаратного обеспечения для параллельного запуска имеющихся программных модулей, мультиверсионное программирование является наиболее подходящим. В отличие от него, схема блока восстановления в общем случае является применимым подходом, который сопоставляется естественным образом с вложенными структурами компонентов и является наиболее подходящим для тех систем, в которых аппаратные ресурсы ограничены, а проверки голосования являются неподходящими [5].

Во многих отношениях мультиверсионное программирование и блоки восстановления взаимно друг друга дополняют с точки зрения их относительных достоинств. Это привело к созданию гибридных схем, которые сочетают в себе различные функции каждого подхода [5]. Например, один или несколько альтернатив из блока восстановления может выполняться параллельно с первичным (при условии, что их вычисления атомарные); издержки времени выполнения альтернативных выполнений будут сведены к минимуму, а их результаты могут быть использованы в приемочных испытаниях. В качестве альтернативы приемочные тесты могут быть использованы в мультиверсионном программировании в помощь алгоритму голосования, и версия может использовать утверждения времени выполнения для обнаружения внутренних ошибок и сигнализировать блоку голосования игнорировать его выход.

Список литературы

1. R. H. Campbell, K. H. Horton and G. C. Belford, "Simulations of a Fault Tolerant Deadline Mechanism", Digest FTCS-9, pp. 95-101, Maddison (WI) (June 1979).
2. P. A. Lee, N. Ghani and K. Heron, "A Recovery Cache for the PDP-II", IEEE Trans. CompUlers, Voi C-29, No 6, pp. 546-549 (1980).
3. Hyland, A Backward Recoverable MC68000 Microcomputer, Final-year undergraduate project, The Hatfield Polytechnic (1985).
4. D. N. Halliwell, An Investigation into the Use of Software Fault Tolerance in a MASCOT-based Naval Command and Control System, Reference A049/DD.17/1, MARL, Newcastle upon Tyne (Feb. 1984).
5. T. Anderson, et al., "Software Fault Tolerance: An Evaluation", IEEE Trans S/W Eng. Vol. SE-II, No. 12 (Dec. 1985).



КОНТРОЛЬ НЕСООТВЕТСТВИЙ РАЗНОРОДНЫХ ТЕХНИЧЕСКИХ ДОКУМЕНТОВ ПРИ РАЗРАБОТКЕ ПЕЧАТНЫХ ПЛАТ

Чернов Ю.А.

научный руководитель канд. техн. наук Носкова Е.Е.

Сибирский федеральный университет

На современных радиоэлектронных производствах все процессы разработки проектной и технологической документации производятся в электронном виде на основе PLM-решений, предполагающих создание общего электронного хранилища информации и организацию электронного технического документооборота.

Проектирование принципиальной электрической схемы осуществляется инженером-схемотехником в САПР печатных плат и, для дальнейшего конструирования печатной платы, информация передаётся инженеру-топологу в виде перечня элементов (ПЭ) в файле формата *.doc.

Основным конструкторским документом сборочного чертежа печатной платы является спецификация, оформляемая в виде таблицы по правилам ГОСТ. При записи в спецификацию составных частей, являющихся элементами электрической принципиальной схемы, в графе «Позиционное обозначение» указано буквенно-цифровое позиционное обозначение элементов. При проектировании электронной аппаратуры принципиальная схема конструктивно выполняется на уровне печатного узла, помещенного в корпус изделия. В исходной постановке задачи спецификация (СП) разрабатывается в машиностроительной САПР инженером, проектирующим корпус изделия, и сохраняется в файл формата *.xls.

Комплектация, заложенная в ПЭ, а так же количество элементов должно с точностью совпадать с комплектацией и количеством элементов, указанном в СП. Конструктор печатных плат вручную проверяет по распечатанным документам соответствие разработанной спецификации (СП) и перечня элементов (ПЭ), что в свою очередь, отражается на сроках выполнения работы. Следовательно, этот процесс необходимо автоматизировать.

Проблема решена созданием программы, считывающей данные из файлов спецификации (СП) и перечня элементов (ПЭ), затем анализирующей и сравнивающей полученные данные.

Разработка программы выполнена на основе среды MS VisualStudio и языка программирования C#. Для чтения данных из файла спецификации (формат файла *.xls) и перечня элементов (формат файла *.doc) использовалась библиотека Microsoft.Office.Interop. Программа анализирует данные, вначале преобразовывая их к удобному для анализа виду, а затем поэлементно сравнивает перечни элементов. В случае расхождения между документами программа выводит список несоответствий, предлагая пользователю выбрать способ исправления.

Результатом работы программы являются:

- Список несоответствий;
- Исправленный файл спецификации.

На рисунке 1 представлена диаграмма зависимостей компонентов, отражающей взаимодействия компонентов программы.

В функциях “WordRead” и “ExcelRead” организовано чтение данных из неоднородных файлов спецификаций в форматах *.doc и *.xls. соответственно Учитывая то, что файлы генерируются в разных конструкторских программах, возникла

необходимость приводить исходные данные к единой символьной кодировке. Эта задача решается при чтении файлов в этих функциях.

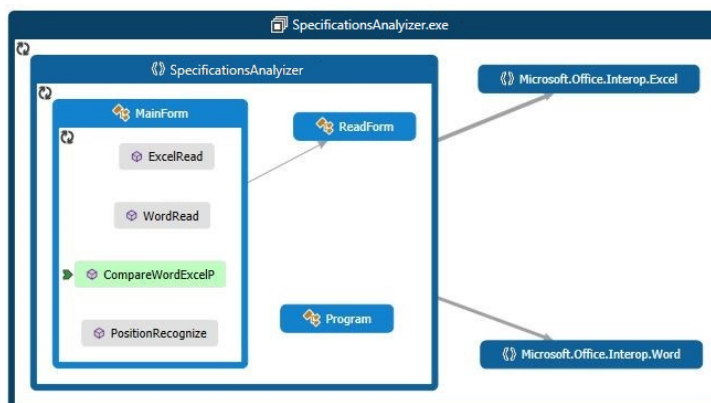


Рис.1 – Диаграмма зависимостей

Для нахождения поэлементных несоответствий был разработан алгоритм распознавания перечислений позиций (вида «R1,R3...R4»и т.п.), к которым относится данный элемент. В результате создана функция “PositionRecognize”, реализующая описанную задачу и приводящая исходные данные в удобный для сравнения вид.

Для сравнения данных была создана функция “CompareWordExcelP”, обнаруживающая расхождения в данных из файлов и определяющая их тип. В результате сравнения функция создает таблицу с найденными несоответствиями. Пример такой таблицы приведен на рисунке 2.

Поз.об.	Наименование	Поз.об.	Наименование
C355	Конденсатор6	C355	Конденсатор15
C357	Конденсатор6	C357	Конденсатор15
C359	Конденсатор6	C359	Конденсатор15
C361	Конденсатор6	C361	Конденсатор15
C364	Конденсатор6	C364	Конденсатор15
HL1	Индикатор2	HL1	Индикатор1
HL2	Индикатор1	HL2	Индикатор2
L2	Индуктивность1	L2	Индуктивность2
L3	Индуктивность3	L3	Индуктивность2
L4	Индуктивность4	L4	Индуктивность3
L5	Индуктивность5	L5	Индуктивность1
R5	Резистор3	R5	Резистор1
R6	Резистор3	R6	Резистор1
R8	Резистор3	R8	Резистор1
R9	Резистор3	R9	Резистор1
R20	Резистор3	R20	Резистор8
R22	Резистор3	R22	Резистор4
R24	Резистор3	R24	Резистор1
R30	Резистор3	R30	Резистор8
R31	Резистор3	R31	Резистор8
R32	Резистор3	R32	Резистор2
R33	Резистор3	R33	Резистор8
C379	Конденсатор1		Элемент не найден в "doc1"
C335	Конденсатор4		Элемент не найден в "doc1"
C372	Конденсатор5		Элемент не найден в "doc1"
C373	Конденсатор5		Элемент не найден в "doc1"
C378	Конденсатор5		Элемент не найден в "doc1"
C383	Конденсатор8		Элемент не найден в "doc1"
C381	Конденсатор14		Элемент не найден в "doc1"
C120	Конденсатор15		Элемент не найден в "doc1"
C148	Конденсатор15		Элемент не найден в "doc1"
C176	Конденсатор15		Элемент не найден в "doc1"
C204	Конденсатор15		Элемент не найден в "doc1"
C251	Конденсатор15		Элемент не найден в "doc1"
C279	Конденсатор15		Элемент не найден в "doc1"
C363	Конденсатор15		Элемент не найден в "doc1"
C366	Конденсатор15		Элемент не найден в "doc1"
C367	Конденсатор15		Элемент не найден в "doc1"
C368	Конденсатор15		Элемент не найден в "doc1"
C369	Конденсатор15		Элемент не найден в "doc1"
C370	Конденсатор15		Элемент не найден в "doc1"
C371	Конденсатор15		Элемент не найден в "doc1"
C382	Конденсатор15		Элемент не найден в "doc1"
VL6	Пыл.2		Элемент не найден в "doc1"

Рис.2 – Список несоответствий

Программа позволяет значительно сократить время анализа двух объемных документов, и с помощью удобного интерфейса пользователя исправить найденные несоответствия.

Список литературы

1.MSDN. Программирование приложений Office (C# и Visual Basic) [Электронный ресурс]. Режим доступа: [://msdn.microsoft.com/ru-ru/library/ee342218.aspx](http://msdn.microsoft.com/ru-ru/library/ee342218.aspx)

О НЕПАРАМЕТРИЧЕСКОМ АЛГОРИТМЕ ДУАЛЬНОГО УПРАВЛЕНИЯ МНОГОМЕРНЫХ БЕЗЫНЕРЦИОННЫХ СИСТЕМ С ЗАПАЗДЫВАНИЕМ

Чжан Е.А.

научный руководитель д-р техн. наук Медведев А.В.

Сибирский федеральный университет

Введение. Впервые алгоритм дуального управления был предложен А.А. Фельдбаумом [1]. Особенность алгоритма состоит в том, что одновременно достигаются две цели – изучение объекта и приведение в заданное состояние [2]. Вся сложность состоит в том, что эти две цели конкурируют между собой. Алгоритмы дуального управления целесообразно применять в двух случаях: небольшой временной горизонт и быстро изменяющиеся характеристики процесса [3]. Изначально алгоритм предназначался для управления систем, чья математическая структура известна с точностью до параметров, так называемые параметрические алгоритмы дуального управления [4]. Однако в большинстве практических задач математическое описание объекта остается неизвестным, в этом случае возможным решением является использование непараметрических методов [5]. На этом пути широкое распространение получили дуальные алгоритмы управления, основанные на нейронных сетях [6].

В работе предлагается непараметрический алгоритм дуального управления многомерными безынерционными системами с запаздыванием.

Постановка задачи. Рассмотрим систему управления, общая схема которой представлена на рис. 1. Математическое описание объекта может иметь следующий вид:

$$x(t) = A(\mu(t), u(t)) + \xi(t), \quad (1)$$

где A – неизвестный оператор объекта.

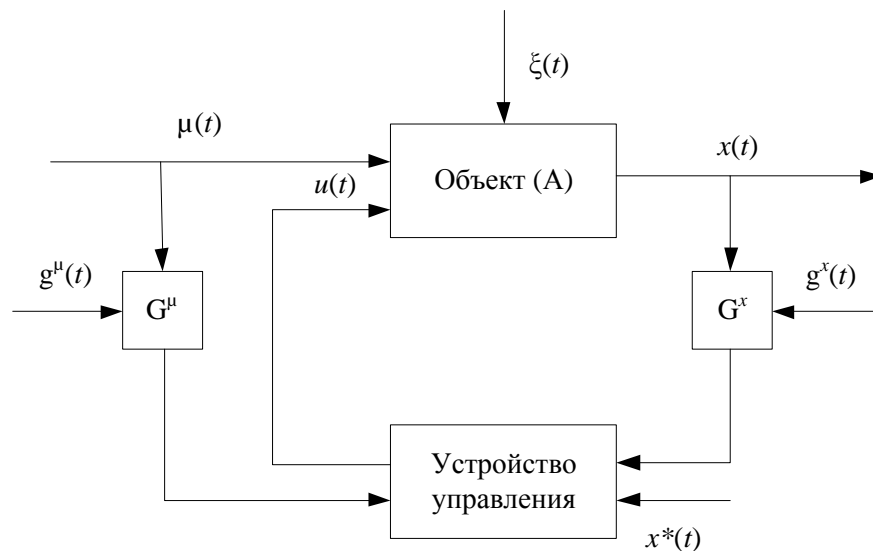


Рис.1 – Замкнутая схема управления

На рисунке 1 приняты следующие обозначения: $\mu(t) = (\mu_1(t), \mu_2(t), \dots, \mu_k(t))$ – неуправляемые входные переменные, $u(t) = (u_1(t), u_2(t), \dots, u_m(t))$ – управляющее воздействие, $\xi(t)$ – возмущающее воздействие, $x(t) = (x_1(t), x_2(t), \dots, x_n(t))$ – выходные переменные процесса, $x^*(t) = (x^*_1(t), x^*_2(t), \dots, x^*_n(t))$ – векторное задающее воздействие. Переменные $\mu(t)$, $x(t)$ измеряются через каналы связи G^μ , G^x со

случайными помехами $g^u(t)$ и $g^x(t)$. Таким образом, измеряя значения входных и выходных переменных процесса с интервалом времени Δt , имеем исходную выборку наблюдений $\{\mu_i, u_i, x_i, i = \overline{1, s}\}$, где s – объем выборки.

В устройстве управления заключен алгоритм, позволяющий сформировать управляющее воздействие $u(t)$ и привести выход объекта $x(t)$ к задающему воздействию $x^*(t)$.

Алгоритм дуального управления. Особенностью предлагаемого алгоритма является то, что не требуется априорная информация о математической структуре исследуемого объекта [5]. Кроме того, алгоритм предназначен для управления многомерными системами, имеющими несколько входных переменных, как управляемых, так и неуправляемых, и несколько выходных.

В многомерном случае на вход объекта поступает векторное управляющее воздействие $u(t) = (u_1(t), u_2(t), \dots, u_m(t))$, которое может быть вычислено в соответствии с предлагаемым непараметрическим алгоритмом дуального управления:

$$u_{j,s+1} = u_{j,s}^* + \Delta u_{j,s+1}, j = \overline{1, m}. \quad (2)$$

где $u_{j,s}^*$ – слагаемое, которое содержит в себе информацию, «знания» об объекте, $\Delta u_{j,s+1}$ – поисковый шаг.

Формула (2) для нахождения управляющего воздействия состоит из двух слагаемых. При этом поисковый шаг для каждой компоненты управляющего воздействия определяется одинаково:

$$\Delta u_{j,s+1} = \sum_{i=1}^n \theta_i (x_{i,s+1}^* - x_{j,s}), j = \overline{1, m}. \quad (3)$$

где $\theta_i \in [0;1]$ – настраиваемые коэффициенты.

Компонента $u_{j,s}^*$ для каждой переменной m -мерного вектора $u(t)$ вычисляется последовательно. Таким образом, алгоритм представляет собой цепочку вычислений. На первом этапе вычисляем $u_{1,s}^*$:

$$u_{j,s}^* = \frac{\sum_{i=1}^s u_{1,i} \prod_{l=1}^k \Phi\left(\frac{\mu_{l,s+1} - \mu_{l,i}}{c_s^\mu}\right) \prod_{l=1}^n \Phi\left(\frac{x_{l,s+1}^* - x_{l,i}}{c_s^x}\right)}{\sum_{i=1}^s \prod_{l=1}^k \Phi\left(\frac{\mu_{l,s+1} - \mu_{l,i}}{c_s^\mu}\right) \prod_{l=1}^n \Phi\left(\frac{x_{l,s+1}^* - x_{l,i}}{c_s^x}\right)}, \quad (4)$$

где $\Phi\left(\frac{\mu_{l,s+1} - \mu_{l,i}}{c_s^\mu}\right)$, $\Phi\left(\frac{x_{l,s+1}^* - x_{l,i}}{c_s^x}\right)$ – ядерные колоколообразные функции и

параметры размытости c_s^μ, c_s^x удовлетворяют условиям сходимости [7, 8].

Затем для каждой последующей переменной $u_j(t), j = \overline{2, m}$ слагаемое $u_{j,s}^*, j = \overline{2, m}$ вычисляется с учетом предыдущего полученного значения $u_j(t), j = \overline{1, m-1}$ с помощью следующей формулы:

$$u_{j,s}^* = \frac{\sum_{i=1}^s u_{j,i} \prod_{l=1}^{j-1} \Phi\left(\frac{u_{l,s+1} - u_{l,i}}{c_s^u}\right) \prod_{l=1}^k \Phi\left(\frac{\mu_{l,s+1} - \mu_{l,i}}{c_s^\mu}\right) \prod_{l=1}^n \Phi\left(\frac{x_{l,s+1}^* - x_{l,i}}{c_s^x}\right)}{\sum_{i=1}^s \prod_{l=1}^{j-1} \Phi\left(\frac{u_{l,s+1} - u_{l,i}}{c_s^u}\right) \prod_{l=1}^k \Phi\left(\frac{\mu_{l,s+1} - \mu_{l,i}}{c_s^\mu}\right) \prod_{l=1}^n \Phi\left(\frac{x_{l,s+1}^* - x_{l,i}}{c_s^x}\right)}. \quad (5)$$

В отличие от предыдущего случая здесь появляется ядерная функция $\Phi\left(\frac{u_{l,s+1} - u_{l,i}}{c_s^u}\right)$ для входной управляющей переменной, а также параметр размытости c_s^u , который также удовлетворяют условиям сходимости [7, 8]. Параметры размытости являются векторными переменными: $c_s^u = (c_s^{u_1}, c_s^{u_2}, \dots, c_s^{u_k})$, $c_s^u = (c_s^{u_1}, c_s^{u_2}, \dots, c_s^{u_m})$, $c_s^x = (c_s^{x_1}, c_s^{x_2}, \dots, c_s^{x_n})$. Настройка данных параметров также осуществляется последовательно [8].

Вычислительный эксперимент. Рассмотрим результаты моделирования объекта, имеющего три входные переменные $u(t) = (u_1(t), u_2(t), u_3(t))$ и две выходные $x(t) = (x_1(t), x_2(t))$. Пусть объект описывается следующей системой уравнений:

$$\begin{cases} x_1(t) = 2 \sin u_1(t) + u_2(t) + \sin u_3(t) + g^{x_1}(t), \\ x_2(t) = 0,4 \sin u_1(t) + u_2^2(t) + 0,5u_3(t) + g^{x_2}(t), \end{cases} \quad (6)$$

где $g^{x_1}(t)$, $g^{x_2}(t)$ – случайные помехи:

$$g^{x_i}(t) = k\phi_i x_i(t), i=1,2, \quad (7)$$

где ϕ_i – случайные величины, распределенные по равномерному закону в интервале $[0;1]$, k – уровень помехи. Так, если помеха составляет 10%, то $k = 0,1$.

Рассмотрим случай комбинированного накопления информации [8]. Была сгенерирована выборка $\{u_{1i}, u_{2i}, u_{3i}, x_i, i = \overline{1,200}\}$. С 201 такта начинает работать алгоритм дуального управления (1). Пусть задающее воздействие имеет вид ступеньки:

$$\begin{cases} x_1^*(t) = 1, & \text{если } 200 < t \leq 450; \\ x_1^*(t) = 1,5, & \text{если } 450 < t \leq 700; \\ x_2^*(t) = 0,15, & \text{если } 200 < t \leq 550; \\ x_2^*(t) = 0,5, & \text{если } 450 < t \leq 700; \end{cases} \quad (8)$$

Рассмотрим результаты вычислительных экспериментов в условиях отсутствия помех. На начальном этапе алгоритм настраивается, а затем приводит выход объекта к заданному значению (13). На рисунке 2 показан график задающего воздействия и выход объекта по переменной $x_1(t)$ и $x_2(t)$. В случае отсутствия помех значение выхода объекта совпадает с величиной задающего воздействия.

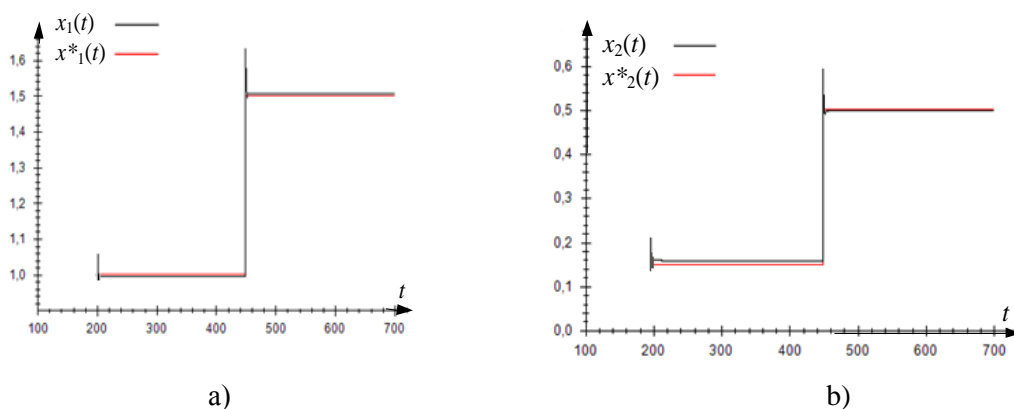
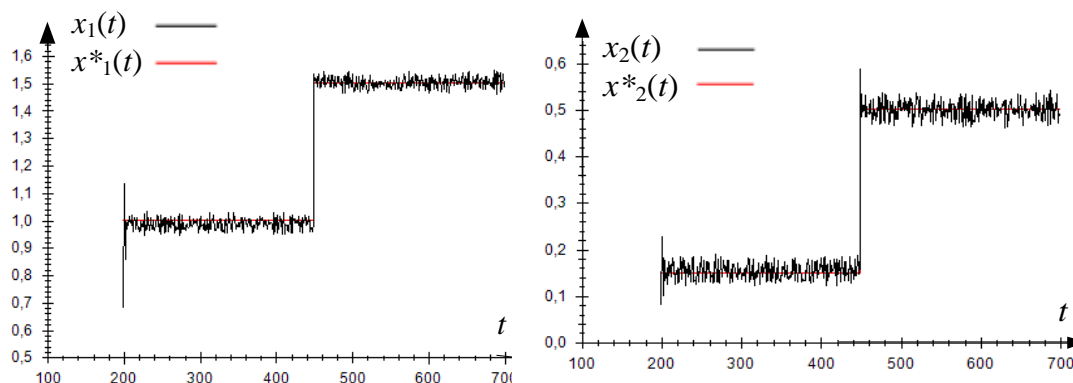


Рис.2 – Результат работы алгоритма управления при ступенчатом задающем воздействии в отсутствии помех: а) для переменной $x_1(t)$, б) для переменной $x_2(t)$

Теперь проведем эксперимент, в ходе которого на выход объекта действовала помеха 5% ($k = 0,05$).



a)

b)

Рис.3 – Результат работы алгоритма управления при ступенчатом задающем воздействии при 5% уровне помех: а) для переменной $x_1(t)$, б) для переменной $x_2(t)$

В этом случае объект также приходит к желаемому заданию, однако вследствие действия помех не точно совпадает с ним, а имеет некоторое отклонение от задающего воздействия.

Заключение. Основной итог настоящей работы состоит в том, что предложен новый непараметрический алгоритм управления многомерным безынерционным объектом, у которого имеется несколько входных и выходных переменных. Особенность данного алгоритма в том, что не требуется априорная информация о математической структуре управляемого объекта. Вычислительные эксперименты показали эффективность предложенного алгоритма при моделировании многомерного нелинейного объекта.

Список литературы

1. Фельдбаум А.А. Основы теории оптимальных автоматических систем. – М.: Физматгиз, 1963. – 552с.
2. Wittenmark B. Adaptive dual control methods: An overview, in Proc. 5th IFAC Symp. Adaptive Syst. Control Signal Processing, 1995, pp. 67–73.
3. К. Åström, В. Wittenmark, Problems of identification and control, J. Math. Anal. Appl., vol. 34, pp. 90–113, 1971.
4. Цыпкин Я.З. Основы теории обучающихся систем / Я.З. Цыпкин. – М.: Наука, 1970. – 252 с.
5. Медведев А.В. Теория непараметрических систем. Управление – I / А.В. Медведев // Вестник Сибирского государственного университета имени академика М.Ф. Решетнева. – 2013. – №2(48). – С.57-63.
6. Fabri S. G., Bugeja M. K., Functional adaptive dual control of a class of nonlinear MIMO systems, Transactions of the Institute of Measurements and Control. 37 (2015) 1009-1025.
7. Надарая Э.А. Непараметрическое оценивание плотности вероятностей и кривой регрессии. – Тбилиси: Издательство Тбилисского университета, 1983. – с.
8. Медведев А.В. Основы теории адаптивных систем. – Красноярск: изд-во Сиб. гос. аэрокосмич. ун-та, 2015, 525 с.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ПОСЛЕДОВАТЕЛЬНО-ИТЕРАЦИОННОГО АЛГОРИТМА РАЗМЕЩЕНИЯ

Щербаков А.О.

научный руководитель канд. техн. наук Носкова Е.Е.

Сибирский федеральный университет

При решении задач размещения и трассировки на этапе конструкторского проектирования электронной аппаратуры (ЭА) проектировщик должен учесть множество требований, предъявляемых к конструкции. Оптимальное размещение элементов обеспечивает повышение надёжности ЭА, уменьшение размеров конструктивных единиц, минимизацию взаимных наводок, задержек сигналов, уменьшение общей длины соединений. Как правило, на радиоэлектронных производствах региона размещение компонентов на печатной плате при проектировании печатных узлов осуществляется ручным, не автоматизированным способом. Поэтому возникает необходимость разработки и внедрения алгоритмов размещения на соответствующих этапах проектирования ЭА, что позволит формализовать требования к конструктивам в виде целевой функции, адаптировать известные алгоритмы конструирования ЭА и разработать уникальные для электронных устройств заданного класса, что сократит время на проектирование электронных узлов, повысит их качество и надёжность.

Задача размещения может быть сформулирована следующим образом. Дано множество элементов $X = \{e_i \mid i = \overline{1, n}\}$, соединённых друг с другом множеством связей $U \subseteq V \times V$. В качестве модели представления данных используется неориентированный граф $G(X, U)$. Информация о связях элемента содержится в матрице смежности графа, которая имеет вид:

$$R = \|r_{i,j}\|, \text{ где } i, j = \overline{1, n}$$

Под областью размещения понимается множество позиций для размещения элементов, организованных в прямоугольную матрицу, имеющих одинаковые линейные размеры и характеризующиеся координатами (s, t) . Необходимо построить размещение $P = \langle p_1, p_2, \dots, p_n \rangle$ такое, что каждой позиции области размещения p_i соответствует некоторый элемент $e_i \in V$, при этом суммарная длина связей элементов минимальна.

Для программной реализации был выбран последовательно-итерационный алгоритм размещения, позволяющий эффективно решать задачу размещения за счёт этапа предварительного размещения [1]. Таким образом, двухэтапное решение задачи размещения позволяет на первом этапе создать благоприятный вариант для итерационного улучшения и сократить время выполнения итерационной части, что очень существенно для схем большой сложности.

Последовательная часть, которая применяется для упорядочивания множества вершин и получения начального размещения, использует понятие коэффициента связности для каждой вершины.

Пусть вершина $x_i \in X$ помещена в $(j+1)$ -ю позицию, где j – число занятых позиций, тогда коэффициентом связности $\Delta(x_i)$ называется выражение:

$$\Delta(x_i) = a_{i,h} - a_{i,z},$$

где $a_{i,h}$ – число ребер, связывающих x_i -ю вершину с ранее размещёнными; $a_{i,z} = \rho(x_i) - a_{i,h}$, т.е. число ребер, связывающих x_i с не размещёнными вершинами, где $\rho(x_i)$ – число ребер, инцидентных вершине x_i (локальная степень x_i). Тогда $\Delta(x_i) = 2a_{i,h} - \rho(x_i)$, где $a_{i,h}$ число ребер, соединяющих x_i с ранее размещёнными вершинами.

Значение $\Delta(x_i)$ определяется для всех не размещенных вершин и выбирается x_i с $\max \Delta(x_i)$. Вершина x_i помещается в первую свободную позицию (узел сетки G_r). Как правило размещение вершин начинается с крайнего левого узла сетки. Процесс последовательно продолжается до тех пор, пока не будут размещены все вершины графа G .

Итерационная часть алгоритма размещения основана на понятии центра тяжести вершины. Для оценки степени «предпочтительности» v -й позиции для каждой вершины графа x_j вводится понятие средней длины ребра – L_j .

$$L_j = \frac{\sum_{i=1}^n d_{i,j} * r_{i,j}}{\sum_{i=1}^n r_{i,j}}, i \neq j,$$

где $d_{i,j}$ – расстояние между узлами сетки, в которые помещены вершины x_i и все смежные ей j -е вершины; $r_{i,j}$ – число связей вершины x_j со всеми смежными с ней j – ми вершинами графа G ; L_j – средняя длина ребер, инцидентных вершине x_j , помещенной в позицию v .

Очевидно, что $(\forall u_{i,j} \in U_j)[(d_{i,j} = 1) \Rightarrow L_j = 1]$, где U_j – множество ребер, инцидентных вершине x_j , помещенной в позицию v . Необходимо стремиться к такому размещению вершин графа, когда средняя длина ребер стремится к минимальному значению. Из множества вершин графа, расположенных в сетке, выбирается вершина x_j , имеющая максимальное значение L_j и производится ее перестановка с другой вершиной с целью минимизации $L(G) = \frac{1}{2} \sum_{i,j} d_{i,j} * r_{i,j}$. Очевидно, что в перестановках должны участвовать только смежные вершины.

Для характеристики системы точек, связанных с вершиной x_j вводится понятие центра тяжести, что позволяет рассматривать их как систему связанных материальных точек и отыскать такое положение для x_j , которое обеспечит равновесие в сетке с точки зрения минимизации $L(G)$. Вычисление координат центра тяжести (s_c и t_c) системы материальных точек производится по формулам:

$$\left\{ \begin{array}{l} s_c = \frac{\sum_{j=1}^n S_j * r_{i,j}}{\rho(x_i)} \\ t_c = \frac{\sum_{j=1}^n t_j * r_{i,j}}{\rho(x_i)} \end{array} \right\}$$

Координаты "центра тяжести" вершины x_j , где $r_{i,j}$ – количество связей между вершиной x_0 и смежными с ней вершинами $\{x_i\}$; вершина x_i – это вершина с максимальным значением L_j ; $\rho(x_i)$ – степень вершины x_i .

Оценка качества решения задачи размещения производится по формуле основного критерия оптимальности – суммарной длины соединений (СДС). :

$$L_{\text{общ}} = \frac{1}{2} \sum_i^N \sum_j^N d_{i,j} * r_{i,j}$$

Данная оценка вычисляется после выполнения каждой итерации алгоритма размещения и ее значение сравнивается со значением, вычисленном на предыдущей итерации. Принятая перестановка вершин должна приводить к уменьшению величины суммарной длины соединений, в противном случае вершины не меняют свое местоположение.

На основании рассмотренного математического аппарата решения задачи размещения последовательно-итерационный алгоритм может быть представлен в последовательности следующих шагов:

1. На заданную монтажную конструкцию наложить координатную сетку с осями координат s, t , т.е. построить граф G_r .
2. Выполнить первоначальное размещение вершин графа схемы G , совместив их с узлами сетки G_r .
3. Для всех вершин графа G вычислить значения суммарных длин связей L_j .
4. Выбрать вершину x_i с max L_j .
5. Вычислить для вершины x_i координаты центра тяжести s_c и t_c .
6. Определить подмножество вершин X' , вершины которого определяют область возможных перестановок с вершиной x_i . Произвести парные перестановки вершин множества X' , с вершиной x_i вычислить для каждой вершины $x_j \in X'$, а также для вершины x_i величины L_j^v и L_j^k с учетом нового местоположения (L_j^v - это длина связей -ой вершины, совмещенной с узлом сетки v).
7. Вычислить отклонение $\sigma_j^v = L_j^v - L_j^k$ для $x_j \in X'$, для всех вершин, выбранных в соответствии с п.п.4,6, где v и k соответственно «старая» и «новая» позиции местоположения вершин.
8. Вычислить значения $\delta_{ij} = \sigma_i^v + \sigma_i^a$ парных перестановок вершин множества X' и вершины x_i . Выбрать такую пару вершин, для которой значение δ_{ij} максимальное.
9. Произвести перестановку вершин.

Программная реализация выполнена на языке C# с использованием графической подсистемы WPF, в среде разработки Visual Studio.

На рисунке 1 приведена схема взаимодействия компонентов.

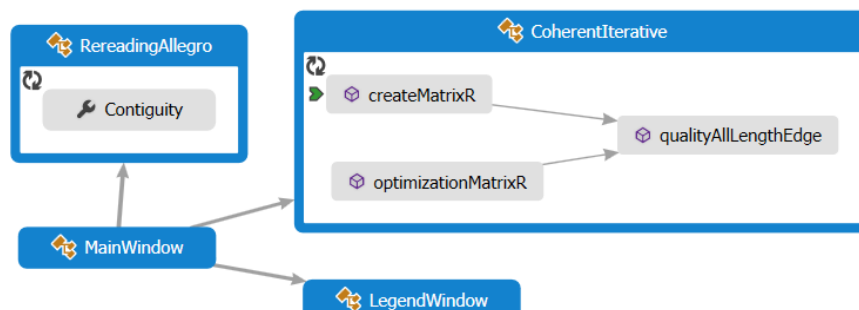


Рис.1 - Схема взаимодействия компонентов

Класс *CoherentIterative* реализует алгоритм формирования коммутационной схемы, имеет два метода *createMatrixR* и *optimizationMatrixR*, реализующие подготовительный и итерационный этапы соответственно. Оценка СДС реализована в методе *qualityAllLenghtEdge*.

Исходными данными для работы алгоритма размещения является матрица соединений элементов, формируемая в программе на основе файла списка соединений, создаваемого в современных САПР печатных плат. Компонент *ReadingAllegro* выполняет считывание файлов формата Allegro и формирует матрицу соединений.

На рисунке 2 приведено главное окно программы.

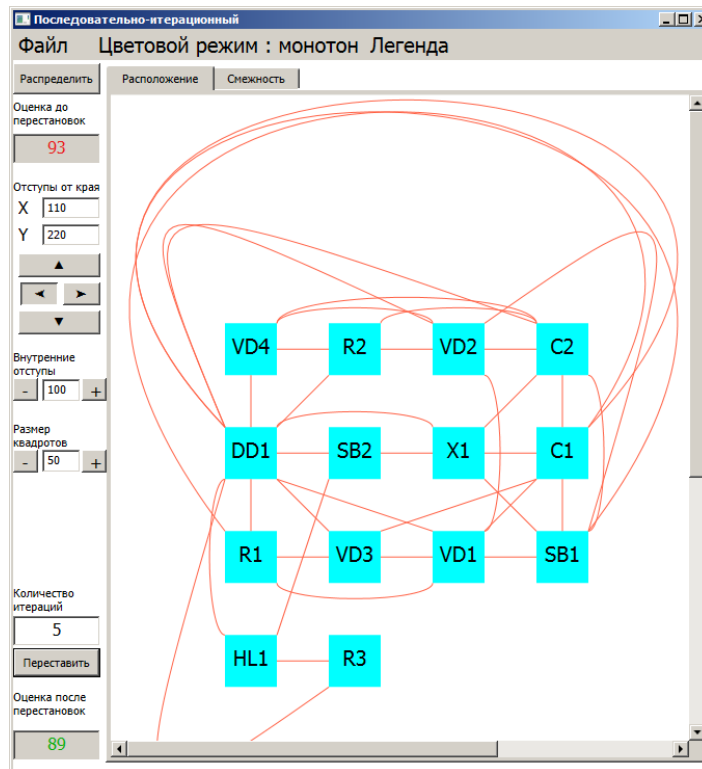


Рис.2 – Главное окно программы

Важными элементами графического интерфейса являются настройки полей ввода входных и управляющих параметров последовательно-итерационного алгоритма размещения, к которым относятся параметры настройки коммутационного поля: отступы от края, внутренние отступы, размер позиций, измеряемых в относительных единицах.

Результатом работы алгоритма размещения является коммутационная схема в виде размещения элементов по позициям с визуализацией соединений. Для улучшения визуального восприятия схемы и минимизации наложения линий на элементы построение линий осуществляется особым образом. Линии различаются типом построения, таким как прямая, эллиптическая дуга и кривая Безье. Линии делятся на несколько групп, что позволяет использовать нужный тип построения: элементы расположенные в одном столбце или строке; элементы расположены в соседних строках или столбцах; элементы находятся у края платы и расположены у соседних границ; элементы находятся у края платы и расположены с противоположных границ; связи не вошедшие ни в одну из групп.

Величина СДС как основной выходной параметр работы программы представлена до и после итерационного этапа.

Список литературы

1. Курейчик, Виктор Михайлович Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для высших учебных заведений./ В.М. Курейчик. В.М. – М.: Радио и связь, 1990. –352 с

КИБЕРНЕТИЧЕСКИЙ ПОДХОД В МОДЕЛИРОВАНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Яршенко Д.И.

научный руководитель канд. техн. наук, доц. Якунин Ю.Ю.

Сибирский федеральный университет

Исследование образовательного процесса в России представляет собой существенный интерес, что обуславливает его актуальность [1]. Как отмечается в [2] в последние десятилетия уровень образования несколько снизился. Тем не менее, необходимо отметить, что и в настоящее время есть студенты, которые имеют большое желание учиться и заниматься научной деятельностью, в итоге они достигают высоких результатов. В этом убеждают многочисленные победы российских студентов на многих международных конкурсах. Настоящий доклад посвящен некоторым аспектам образовательного процесса. Одним из важных направлений на этом пути является создание компьютерного «портрета» образовательного процесса и технологии работы с ним. Под компьютерным «портретом» понимается комплекс алгоритмов и компьютерная программа, производящая математические вычисления, соответствующие образовательному процессу, с целью его исследования и усовершенствования. Компьютерный «портрет» позволял бы прогнозировать основные параметры или показатели образования выпускников университетов при соответствующих управляемых и неуправляемых воздействиях на образовательный процесс. Такое исследование образовательного процесса необходимо начинать с формирования входных и выходных характеристик, возмущающих воздействий [3]. В реальной жизни объекты чаще характеризуются набором качественных характеристик. Для образовательного процесса следует формирование таких характеристик, которые будут определять течение исследуемого процесса во времени. Важнейшая роль для дальнейшего будет принадлежать этапу формированию вектора входных и выходных переменных, входящих в разрабатываемую модель образовательного процесса.

Для математического моделирования образовательного процесса [4] определимся с достаточно общей его структурой. На рисунке 1 представлено первоначальное видение общей схемы для моделирования образовательного процесса. Как видно из рисунка, образовательный процесс включает в себя объект и субъект управления. Стрелки, помещенные внутри объекта и субъекта образовательного процесса, означают присутствие человека или коллектива людей. На рисунке использованы следующие обозначения: $\mu(t)$ – векторная входная переменная процесса; $\xi(t)$ – векторное случайное воздействие (психологический, эмоциональные факторы, присущие человеку); $\nu(t)$ – внешние воздействия, которые зависят от разных факторов и имеют самый разный характер (региональный, экономико-социальный и др.); $u(t)$ – управляющее воздействие, которое находится внутри организационной системы – университет; $x(t)$ – промежуточные показатели, которые объединяют в группы те или иные входные переменные $\mu(t)$; $y(t) = y(z_1(t), z_2(t))$ – векторная выходная переменная процесса, которая измеряет как выход субъекта, так и выход объекта управления; α, β и γ, ρ – некоторые коэффициенты, нужные для оценки параметров (определяться они будут на основе экспериментальных данных, либо будут задаваться исследователем). Также на организационную систему влияет внешнее воздействие – Министерство образования и науки РФ.

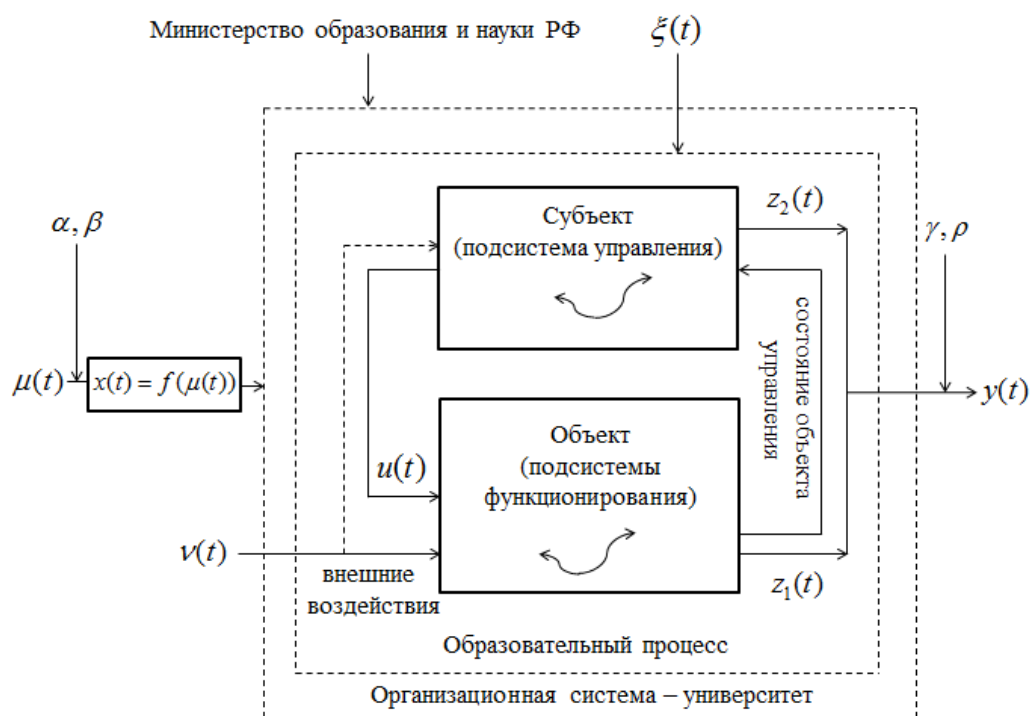


Рис.1 - Общая схема структуры образовательного процесса

Для того чтобы определиться с первоначальными входными и выходными переменными в предшествующие годы, с учетом показателей, рекомендованных Министерством образования и науки Российской Федерации, был проведен достаточно обширный анкетный опрос респондентов (студентов и преподавателей). На основании проведенного опроса были сформированы наборы переменных, влияющих на образование: $\mu = (\mu_1, \mu_2, \dots, \mu_m)$ – факторы (показатели), влияющие на образование, где $k = \overline{1, m}$; $x = (x_1, x_2, \dots, x_n)$ – промежуточные показатели качества образования, где $i = \overline{1, n}$; $y = (y_1, y_2, \dots, y_s)$ – итоговые показатели качества образования, где $j = \overline{1, s}$ [3].

С учетом ретроспективного анализа были выделены показатели для советского и современного периодов, представляющие собой вектор входных воздействий: μ_2 – учебные материалы; μ_4 – баллы аттестата; μ_5 – доступность связи с преподавателем, под данным фактором понимается взаимоотношения преподавателя со студентами, способность понятно преподнести предмет студентам и др.; μ_6 – мотивация для учебы; μ_7 – бытовые условия; μ_8 – работа с научным руководителем; μ_{10} – желание учиться; μ_{11} – чтение русской литературы и истории; μ_{12} – статус преподавателя, предполагает наличие ученой степени у преподавателя, его статус в университете; μ_{13} – отчетность; μ_{14} – заработная плата ППС; μ_{16} – уровень востребованности выпускников; μ_{20} – статус студента в современном обществе. Также были выделены показатели, отдельно влияющие на образование советского и нынешнего периодов. Набор показателей, характерных только для советского периода: μ_{17} – вступительные экзамены в университет; μ_{19} – оборудование для лабораторных работ. Набор показателей современного периода: μ_1 – оснащение техникой; μ_3 – результаты ЕГЭ; μ_9 – комфортные условия для занятий; μ_{15} – международная деятельность.

Промежуточные показатели качества образовательного процесса для современного периода: x_1 – результаты экзаменов, зачетов; x_2 – научные результаты (статьи, диссертации); x_3 – стипендии, гранты; x_4 – коммуникабельность. Для советского периода: x_1 – результаты экзаменов, зачетов; x_2 – стипендии, гранты; x_3 – коммуникабельность.

В качестве итоговых показателей качества y_s для обоих периодов были выделены: y_1 – знания в профессиональной сфере и y_2 – кругозор (широта знаний).

Следует отметить, что были сформированы начальные зависимости промежуточных показателей качества образования от факторов, влияющих на образование. Зависимость промежуточных показателей от факторов представлена следующим образом: $x_i(\mu) = f(\mu_k^{(n)})$, $k = \overline{1, m}$, где $\mu_k^{(n)}$ – составной вектор, который представляет собой набор переменных μ_k , влияющих на тот или иной промежуточный показатель x_i . Определим зависимости для современного периода:

$$\begin{aligned} x_1 &= f_1(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6, \mu_7, \mu_9, \mu_{10}, \mu_{11}); & x_2 &= f_2(\mu_1, \mu_2, \mu_6, \mu_7, \mu_8, \mu_{10}, \mu_{11}); \\ x_3 &= f_3(\mu_2, \mu_5, \mu_6, \mu_7, \mu_8, \mu_{10}); & x_4 &= f_4(\mu_2, \mu_5, \mu_6, \mu_7, \mu_8, \mu_{10}, \mu_{11}). \end{aligned}$$

Определим зависимости для советского периода: $x_1 = f_1(\mu_2, \mu_5, \mu_6, \mu_7, \mu_{10}, \mu_{11}, \mu_{18}, \mu_{20}, \mu_{17})$; $x_2 = f_2(\mu_2, \mu_5, \mu_7, \mu_8, \mu_{10}, \mu_{11}, \mu_{17}, \mu_{18}, \mu_{19}, \mu_{20})$; $x_3 = f_3(\mu_2, \mu_4, \mu_5, \mu_8, \mu_{10}, \mu_{11}, \mu_{12}, \mu_{18})$.

Также были сформированы зависимости итоговых показателей качества $y = (y_1, y_2)$ от промежуточных показателей и от факторов. Зависимость итоговых показателей качества может быть представлена в виде: $y_j(x) = F(x_i^{(s)}, \mu_k^{(s)})$, $j = \overline{1, 2}$. Получаем следующие итоговые показатели качества для современного периода: $y_1 = F(x_1, x_2, x_4)$ и $y_2 = F(x_1, x_2, x_3, x_4)$; для советского периода: $y_1 = F(x_1, x_2)$ и $y_2 = F(x_1, x_2, x_3)$.

На первоначальном этапе данного исследования для образовательного процесса входными характеристиками будут являться промежуточные показатели и факторы, упомянутые выше, а выходными характеристиками – итоговые показатели качества. Компьютерное исследование образовательного процесса поможет получить отклик данной активной системы на соответствующие изменения входных переменных.

Проведение подобной работы предполагает участие в ней специалистов различных профессий, таких как педагогов, психологов и, возможно, опроса тех людей, которые учились еще в советское время. Достигнутый успех в данном направлении может помочь в принятии тех управленческих решений, которые бы привели систему к желаемому состоянию.

Список литературы

1. Медведев А.В. Некоторые замечания об образовательном процессе // Проблемы управления в социальных системах. Изд. Том 8, Вып. 12, 2015. С. 151 – 171.
2. Дискуссия об образовании // Вестник СибГАУ. – 2010, № 5. – С. 197 – 206.
3. Медведев А.В., Ярлыкова Л.К. О компьютерном «портрете» образовательного процесса // Труды IX Всероссийской научно-практической конференции «Системы автоматизации в образовании, науке и производстве (AS'2013)». – Новокузнецк, 2013. – С. 12 – 19.
4. Граничина О.А. Математические модели контроля качества образовательного процесса в учреждениях высшего профессионального образования. – СПб.: Издательство ВВМ, 2008. – 180 с.