

***СЕКЦИЯ 7.***  
***MATLAB в образовании и Интернете***

***Председатель:***  
**к. т. н., доцент С. П. Иглин**

## Оглавление

Алпатов Б. А., Балашов О. Е., Бохан К. А., Катаев А. А., Муравьев В. С., Селяев А. А., Стротов В. В. РАЗРАБОТКА ЦИКЛА ЛАБОРАТОРНЫХ РАБОТ ПО КУРСУ «ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ» В СИСТЕМЕ MATLAB .....	1825
Деменков Н. П. МЕТОДЫ КЛАССИЧЕСКОЙ И СОВРЕМЕННОЙ ТЕОРИИ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ В ЛАБОРАТОРНЫХ РАБОТАХ.....	1826
Добротин С. А., Прокопчук Е. Л. ИСПОЛЬЗОВАНИЕ МАТЕМАТИЧЕСКОГО ПАКЕТА MATLAB/SIMULINK В КУРСЕ МОДЕЛИРОВАНИЯ СИСТЕМ УПРАВЛЕНИЯ.....	1833
Дьяконов В. П. СМОЛЕНСКАЯ ШКОЛА ВНЕДРЕНИЯ И ПОПУЛЯРИЗАЦИИ СИСТЕМЫ MATLAB .....	1838
Камышников В. А. ДЕЛОВЫЕ ИГРЫ В СРЕДЕ MATLAB .....	1842
Козлов П. В., Кучеренко Н. Л. ПРИМЕНЕНИЕ СИСТЕМЫ MATLAB В ПОСТАНОВКЕ КОМПЬЮТЕРНЫХ ЛАБОРАТОРНЫХ ПРАКТИКУМОВ ПО ФИЗИКЕ .....	1849
Коломийцева С. В. MATLAB В КУРСЕ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ НА ЭВМ.....	1852
Костиков П. А., Мулин П. В. ПРИМЕНЕНИЕ MATLAB ПРИ ИЗУЧЕНИИ ПРОФИЛЬНЫХ ДИСЦИПЛИН ДЛЯ СПЕЦИАЛИСТОВ ПО СИСТЕМАМ УПРАВЛЕНИЯ.....	1857
Котельников И. А., Черкасский В. С. АДАПТАЦИЯ ПРИЛОЖЕНИЙ MATLAB ДЛЯ MATLAB WEB SERVER.....	1859
Коткин Г. Л., Черкасский В. С., Котельников И. А. ОПЫТ ИСПОЛЬЗОВАНИЯ СРЕДЫ БЫСТРОЙ РАЗРАБОТКИ ПРИЛОЖЕНИЙ MATLAB В УЧЕБНОМ ПРОЦЕССЕ НА ФИЗИЧЕСКОМ ФАКУЛЬТЕТЕ НГУ.....	1879
Кудашов А. В., Медведева С. Н., Михотин В. Д., Пискарев С. П. MATLAB В УЧЕБНЫХ КУРСАХ ПО ИССЛЕДОВАНИЮ ПЕРЕХОДНЫХ ПРОЦЕССОВ В ЭЛЕКТРОЭНЕРГЕТИЧЕСКИХ СИСТЕМАХ.....	1892

Мулин П. В., Костиков П. А. АНИМАЦИОННАЯ ВИЗУАЛИЗАЦИЯ РЕЗУЛЬТАТОВ МОДЕЛИРОВАНИЯ УПРАВЛЕНИЯ ПОЛЕТОМ В СРЕДЕ MATLAB .....	1915
Одинцов В. И., Конрадов А. А., Кукса Ю. И. ПРИМЕНЕНИЕ MATLAB WEB SERVER В ГЕОФИЗИКЕ ДЛЯ ИНТЕРАКТИВНОЙ АДАПТИВНОЙ ОБРАБОТКИ ДАННЫХ, РАСПРЕДЕЛЕННЫХ В СЕТИ ИНТЕРНЕТ .....	1917
Резванов Д. Х. ПРИМЕНЕНИЕ ИНТЕРАКТИВНОГО ИНСТРУМЕНТАРИЯ FILTER DESIGN&ANALYSIS TOOL В ЛАБОРАТОРНОМ ПРАКТИКУМЕ .....	1931
Стрелковский Н. Н. ОПЫТ ИСПОЛЬЗОВАНИЯ MATLAB ДЛЯ РАСЧЕТА ПАРАМЕТРОВ ПИД-РЕГУЛЯТОРА .....	1933
Тарасевич Ю. Ю., Пономарева И. С. КОМПЬЮТЕРНЫЙ ПРАКТИКУМ ПО КУРСУ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ.....	1935
Хохлов Ю. И., Гельман М. В., Преображенский К. А., Федюнин К. П. КОМПЛЕКС ВИРТУАЛЬНЫХ ЛАБОРАТОРНЫХ РАБОТ ПО ЭЛЕКТРОСНАБЖЕНИЮ И ЭНЕРГЕТИЧЕСКОЙ ЭЛЕКТРОНИКЕ.....	1940
Чернецова Е. А. ОРГАНИЗАЦИЯ ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКИ СТУДЕНТОВ С ИСПОЛЬЗОВАНИЕМ ВОЗМОЖНОСТЕЙ ПАКЕТА MATLAB .....	1945
Шатров М. Г., Черняк Б. Я. РАЗВИТИЕ И ПРИМЕНЕНИЕ НОВЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В УЧЕБНОЙ И ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ С ИСПОЛЬЗОВАНИЕМ ПАКЕТА MATLAB И ЕГО ПРИЛОЖЕНИЙ.....	1951

УДК 621.391.26

## **РАЗРАБОТКА ЦИКЛА ЛАБОРАТОРНЫХ РАБОТ ПО КУРСУ «ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ» В СИСТЕМЕ MATLAB**

*Алпатов Б. А., Балашов О. Е., Бохан К. А., Катаев А. А.,  
Муравьев В. С., Селяев А. А., Стротов В. В.  
Рязанская государственная радиотехническая академия, Рязань  
e-mail: atr\_art@rgrta.ryazan.ru*

Курс «Цифровая обработка изображений» имеет ярко выраженную прикладную направленность. Вследствие этого, совершенство практической части данного курса является необходимым условием успешного изучения методов и алгоритмов обработки изображений. Эта область знаний является быстро и динамично развивающейся, поэтому был разработан новый цикл лабораторных работ, дающих студентам начальные навыки цифровой обработки изображений и охватывающие такие ее разделы, как фильтрация, сегментация, улучшение и корреляционный анализ изображений.

При создании цикла работ были использованы возможности, предоставляемые Image Processing Toolbox. Часть работ ориентирована на знакомство обучаемых с возможностями данного пакета и предполагает самостоятельное написание несложных программ, использующих набор базовых функций обработки изображений. Остальные работы предполагают углубленное изучение студентами определенного раздела курса, а потому для повышения наглядности эти работы выполнены с широким использованием элементов GUI, что позволяет студентам концентрироваться на изучении методов и алгоритмов, не отвлекаясь на тонкости программной реализации.

При постановке данного цикла работ использовались научно-технические результаты, полученные в результате выполнения коллективом авторов ряда научно-исследовательских работ. Это позволило учитывать современные тенденции и использовать последние достижения в области обработки изображений.

УДК 621.3.078

## **МЕТОДЫ КЛАССИЧЕСКОЙ И СОВРЕМЕННОЙ ТЕОРИИ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ В ЛАБОРАТОРНЫХ РАБОТАХ**

*Деменков Н. П.*

*Московский государственный технический университет им. Н. Э. Баумана, Москва,  
e-mail: demenkov@iu1.bmstu.ru*

За последние десятилетия арсенал методов теории управления значительно изменился и пополнился. В недавнем прошлом основным инженерным инструментом был частотный метод, базирующийся на построении логарифмических частотных характеристик. Аналитические соотношения для задач анализа и синтеза систем управления дополнялись графическими построениями и могли быть выполнены вручную.

К настоящему времени и сами объекты управления и их описание существенно усложнились из-за повышения порядка описывающих их уравнений, увеличения размерностей как регулируемых координат, так и управляющих воздействий, появления в контуре управления вычислительных устройств и т. п. Наряду с привычным описанием «вход-выход» получили широкое распространение уравнения состояния динамических систем. С применением последних в практику управления прочно вошли понятия управляемости, наблюдаемости, стабилизируемости, достижимости, идентифицируемости и т. п. [1].

Описание задач идентификации, фильтрации, оптимального управления, базирующееся на описании динамических систем уравнениями состояния, привело к усложнению математического аппарата, широкому использованию матричных соотношений и, как следствие, жесткой необходимости поддержки инженерных расчетов вычислительными и программными средствами. Другими словами, существенное обновление к настоящему времени расчетного инструментария потребовало адекватного отображения в учебном процессе.

Настоящий доклад как раз и призван проиллюстрировать применение в учебном процессе современных методов и программных продуктов в практике регулирования и управления.

Реализация алгоритмов управления, как правило, осуществляется в рамках пакетов прикладных программ [2]. Строго говоря, основные расчетные соотношения могут быть реализованы в различных программных продуктах (СС, SIAM, MBTU и др.), но на сегодняшний день наиболее развитой и широко распространенной как в нашей стране, так и за рубежом, является интегрированная среда MATLAB с огромным набором ин-

струментальных средств. Все излагаемые в учебных курсах по теории управления методы оформлены в этой среде в виде стандартных процедур. Простые языковые средства позволяют выстраивать эти процедуры для решения любых методических задач, начиная с задач формализации и оценки устойчивости и заканчивая исследованием особенностей функционирования искусственных нейронных сетей и их применением для задач прогнозирования и идентификации [3,4].

Пять лабораторных работ посвящены формализации математического описания систем управления [5].

При проектировании САУ используется большое число методов анализа и синтеза. Все они предполагают, что исходная задача описана в математическом виде — формализована. Формализация задачи управления есть первый и важнейший этап ее решения. От выбора системы показателей модели и методов существенно зависит и вид будущей САУ и ее характеристики. Выбор модели в значительной мере зависит от целей проведения исследования и наличия методов с соответствующими свойствами.

При использовании метода переменных вход-выход (ПФ и ЧХ) модель САУ удобно представлять в виде набора типовых динамических звеньев. Типовое динамическое звено — это элемент САУ, имеющий соответствующую передаточную функцию, порядка не выше второго относительно комплексной переменной  $s$ .

В результате выполнения первой ЛР студент получает навыки работы в пакете MATLAB, умение: определить, из каких типовых звеньев состоит исследуемая САУ, сформировать структурную схему САУ из отдельных типовых звеньев, построить ЧХ и переходный процесс отдельных звеньев и всей системы в целом,

Реальные системы автоматического регулирования не являются чисто линейными, и в ряде случаев их поведение не может быть даже приближенно описано линейными дифференциальными уравнениями, т. е. системы не могут быть представлены как линеаризованные.

В ряде случаев специальное применение нелинейных элементов в схемах управления значительно расширяет возможности повышения качества переходных процессов в системе. Примерами таких нелинейных элементов являются нелинейные непрерывные и релейные усилители, функциональные преобразователи и т. п. Введение релейных усилителей, например, в систему управления может перевести ее в режим малых автоколебаний, существование которых повышает ее быстродействие и уменьшает перерегулирование при скачкообразных воздействиях.

В системах автоматического управления встречаются нелинейности самого различного вида. Нелинейные звенья можно классифицировать по различным показателям: симметрии, гладкости, однозначности характеристик и другим признакам.

Основные задачи исследования нелинейных систем сводятся к отысканию возможных состояний равновесия системы и исследованию их устойчивости, исследованию процессов перехода системы к тому или иному установившемуся состоянию при различных начальных отклонениях.

В результате выполнения второй ЛР студент получает навыки по включению нелинейных звеньев в структурную схему САУ с помощью пакета Simulink, умение построить переходный процесс и фазовый портрет нелинейной САУ при различных начальных условиях, оценить влияние параметров нелинейности на характеристики переходного процесса.

Третья ЛР посвящена изучению способов перехода от уравнений типа «вход-выход» к уравнениям в форме Коши (пространства состояний), составлению векторных структурных схем и моделированию (решению) уравнений пространства состояний с помощью MATLAB. В результате ее выполнения студент получает навыки по преобразованию линейных последовательных и параллельных звеньев на структурной схеме САУ, умение работы в пакете Simulink по формированию структурной схемы САУ, построению переходных процессов.

Четвертая ЛР посвящена построению структурных схем замкнутых САР на основе типовых звеньев и изучению их динамических свойств при подаче на вход единичного ступенчатого сигнала. В результате выполнения ЛР студент получает навыки по преобразованию линейных последовательных и параллельных звеньев на структурной схеме САУ при работе в пакете Simulink, умение построения переходных процессов и частотных характеристик системы.

Пятая ЛР поможет овладеть навыками и умением по исследованию свойств систем с единичной, неединичной, жесткой и гибкой обратными связями с использованием средств графического пользовательского интерфейса (GUI).

Принцип управления «по отклонению» имеет ряд преимуществ в сравнении с другими принципами (например, с принципом «компенсации»), однако из-за замкнутости контура система может стать сильно колебательной и даже потерять устойчивость.

В результате выполнения ЛР студент получает навыки работы в графической среде пользователя LTI-Viewer при анализе замкнутой САУ, умение построить и исследовать свойства систем с единичной, неединичной, жесткой и гибкой обратной связью, оценить влияние погрешности изготовления различных элементов системы с обратной связью на точность ее работы.

Четыре лабораторные работы знакомят студентов с проблемами устойчивости систем.

6-я ЛР посвящена овладению навыками и умением по исследованию устойчивости САУ на основе 1-го метода А.М.Ляпунова. В результате

выполнения ЛР студент получает навыки и умения линеаризации дифференциальных уравнений, описывающих САУ, найти корни характеристического уравнения, сделать вывод об устойчивости системы на основе 1-го метода Ляпунова.

В 7-й ЛР изучаются динамические свойства замкнутых линейных систем автоматического регулирования, влияние параметров системы на устойчивость с использованием средств графического пользовательского интерфейса (GUI). В результате выполнения ЛР студент получает навыки работы в графической среде пользователя LTI-Viewer при анализе устойчивости замкнутой САУ, умение построить и исследовать устойчивость линейных систем с помощью алгебраических и частотных критериев устойчивости, исследовать влияние параметров различных элементов системы на ее устойчивость.

8-я ЛР позволяет овладеть навыками и умением по нахождению параметров автоколебаний в нелинейных системах методом гармонической линеаризации. В результате выполнения ЛР студент получает навыки работы в пакете Simulink по формированию фазового портрета замкнутой САУ, построению переходных процессов, умение найти параметры автоколебаний систем и исследовать влияние параметров системы (коэффициента усиления и постоянных времени) на амплитуду и частоту автоколебаний.

9-я ЛР посвящена построению корневого годографа замкнутых САУ и изучению их динамических свойств. В результате выполнения ЛР студент получает навыки по преобразованию линейных последовательных и параллельных звеньев, используя стандартные функции MATLAB, умение построить корневой годограф системы.

Восемь лабораторных работ посвящены изучению методов решения двух центральных проблем теории автоматического управления: проблемы синтеза систем управления по заданным показателям качества и проблемы оптимального выбора такого закона управления, который, удовлетворяя четко заданной системе ограничений, обеспечивает при своей реализации оптимальное значение того или иного показателя качества работы системы.

10-я ЛР посвящена исследованию динамических свойств систем модального управления, который весьма прост в использовании. В результате выполнения ЛР студент получает умение в исследовании влияния корней, выбранных в качестве желаемых, на динамические свойства и параметры системы.

11-я ЛР направлена на овладение навыками и умением по синтезу стабилизирующего управления на основе прямого метода А.М.Ляпунова. В результате выполнения ЛР студент получает навыки и умение сформировать контур управления на основе использования функции Ляпунова,



исследовать влияние параметров системы (коэффициента усиления) на поведение системы.

12-я ЛР направлена на овладение навыками и умением по исследованию скользящих режимов в системах управления. В результате выполнения ЛР студент получает навыки по исследованию скользящих режимов в системах управления и умение определить условия возникновения скользящего режима.

13-я ЛР посвящена овладению навыками и умением по синтезу линейных систем управления с квадратичным критерием качества на основе классического вариационного исчисления. В результате выполнения ЛР студент получает умение по определению оптимального управления методом классического вариационного исчисления, решить задачу аналитического конструирования регулятора, навыки по выбору постоянных весовых коэффициентов в функционале качества.

14-я ЛР посвящена синтезу регулятора методом  $H^\infty$ -оптимизации для непрерывных систем. Для синтеза неустойчивого объекта используется 2-Риккати подход. В результате выполнения ЛР студент получает навыки и умения по изучению динамических особенностей систем, построенных методами  $H^\infty$ -оптимизации

15-я посвящена овладению навыками и умением по синтезу параметров ПИД-регулятора для нелинейных систем управления. В результате выполнения ЛР студент получает навыки по настройке оптимальных коэффициентов в ПИД-регуляторе, умение выбрать оптимальные значения коэффициентов ПИД-регулятора.

16-я ЛР поможет овладеть навыками и умением по построению оптимальных систем на основе принципа максимума Л.С. Понтрягина. В результате выполнения ЛР студент получает навыки по оптимизации систем управления на основе принципа максимума Л.С.Понтрягина, умение определить условия управляемости и достижимости системы, решить краевую задачу методом последовательных приближений.

17-я ЛР посвящена ознакомлению с принципом оптимальности Р. Беллмана. В результате выполнения ЛР студент получает навыки по оптимизации систем управления на основе принципа оптимальности Р.Беллмана, умение определить условия управляемости и достижимости системы, решить задачу оптимизации методом динамического программирования.

Системы автоматического управления, как правило, работают в условиях помех. С методикой исследования систем, находящихся под воздействием случайных сигналов, знакомят студентов три лабораторные работы.

18-я ЛР направлена на приобретение практических навыков для оценки влияния случайных факторов в системах управления по результатам эксперимента. В результате выполнения ЛР студент получает навыки

и умения по экспериментальному определению статистических характеристик в системах управления методом статистических испытаний и методом стохастической аппроксимации.

19-я ЛР посвящена исследованию особенностей цифровых систем управления, обусловленных процессами квантования по времени и по уровню. В результате выполнения ЛР студент получает умение исследовать изменения динамических свойств цифровых систем по сравнению с соответствующими непрерывными системами.

20-я ЛР посвящена овладению навыками и умением по построению наблюдающих устройств на основе фильтра Калмана, определению условий наблюдаемости системы.

Проблема идентификации является одной из основных проблем теории и практики управления. Без знания оператора объекта управления, ставящего в соответствие входные и выходные сигналы, нельзя эффективно решать задачи расчета и проектирования систем управления. Две лабораторные работы знакомят с методами идентификации на основе нейронных систем.

21-я ЛР посвящена изучению принципов построения нечетких логических регуляторов (НЛР) для АСУ ТП. НЛР используются для управления плохо-формализованными объектами и процессами, построение точных математических моделей которых является трудной задачей. Широкое распространение получили методы, позволяющие строить не сами модели объектов, а модели управления этими объектами. При этом моделируется не сам объект, а качественное описание процесса управления этим объектом. Стратегия управления может быть сформулирована как набор правил, использующих нечеткие понятия. Такие правила называются лингвистическими правилами управления (ЛПУ). Нечеткая модель регулятора описывается с помощью набора ЛПУ. Знания модели управления в виде ЛПУ НЛР используют при функционировании. В результате выполнения ЛР студент получает навыки работы с fuzzy-библиотекой пакета MATLAB и умение спроектировать нечеткий логический регулятор, выбрать функции принадлежности, составить правила нечеткой логики.

22-я ЛР посвящена исследованию особенностей функционирования искусственных нейронных сетей и их применению для задач прогнозирования и идентификации. В результате выполнения ЛР студент получает навыки и умения по исследованию особенностей обучения и функционирования искусственных нейронных сетей, возможностей их применения в качестве модельных структур при реализации процедуры идентификации динамических объектов.

Невозможно дать окончательные рецепты для решения широкого спектра задач, порожденных практикой проектирования САУ. Изложенный в докладе материал призван служить лишь базой, позволяющей с

большой скоростью и эффективностью строить новые лабораторные работы для решения задач управления, возникающих на практике.

### ***Литература***

1. Методы классической и современной теории автоматического управления: Учебник в 3 томах / Под ред. Н. Д. Егупова.— М.: Изд-во МГТУ им. Н. Э. Баумана, 2000.
2. *Зверев В. Ю., Фаронов В. В.* Десять лабораторных работ в СиАМ: Метод. указания к лабораторным работам по курсу «Управление в технических системах».— М.: Изд-во МГТУ им. Н. Э. Баумана, 1993.— 52 с.
3. *Медведев В. С., Потемкин В. Г.* Control System Toolbox. MATLAB 5 для студентов / Под общ. ред. к. т. н. В. Г. Потемкина. —М.: Диалог-МИФИ, 1999.— 278 с.
4. Сборник лабораторных работ по курсу «Управление в технических системах»: Методические указания к лабораторным работам / Под ред. К.А.Пупкова.— М.: Изд-во МГТУ им. Н. Э. Баумана, 2002.— 72 с.
5. *Деменков Н. П., Задорожная Н. М.* Математическое описание систем управления: Метод. указания к лаб. работам.— М.: Изд-во МГТУ им. Н. Э. Баумана, 1992.— 24 с.

УДК 681.5. 015

## **ИСПОЛЬЗОВАНИЕ МАТЕМАТИЧЕСКОГО ПАКЕТА MATLAB/SIMULINK В КУРСЕ МОДЕЛИРОВАНИЯ СИСТЕМ УПРАВЛЕНИЯ**

*Добротин С. А., Прокопчук Е. Л.*

*Дзержинский филиал Нижегородского государственного  
технического университета, Дзержинск,  
e-mail: llp@hotmail.ru*

В связи с широким применением вычислительной техники в качестве управляющей в схемах автоматизации производственных процессов возрастают требования к специалистам в данной области, которые согласно «Государственному образовательному стандарту высшего профессионального образования» [2] должны знать:

- аналитические и численные методы для анализа математических моделей технических систем, технологических процессов с использованием компьютерной техники;
- методы математического моделирования и автоматизированного проектирования при разработке и совершенствовании программно-технических средств и систем автоматизации и управления;
- методы разработки алгоритмического и программного обеспечения систем автоматизации и управления.

На протяжении нескольких лет в курсе «Моделирование систем управления» в лабораторных работах использовался язык Fortran. При таком подходе основное внимание уделялось программированию, а не моделированию. Так как анализ поведения многих систем и устройств в динамике обычно базируется на решении систем обыкновенных дифференциальных уравнений, много времени затрачивалось на реализацию алгоритмов решения дифференциальных уравнений, а также на отладку программ.

Наиболее эффективными для моделирования систем управления динамических объектов являются математические пакеты, такие как MathCad и MATLAB. Применение данных пакетов дает возможность наибольшее внимание уделять моделированию и снижает трудоемкость выполнения поставленной задачи.

В настоящее время лабораторные работы проводятся с использованием математического пакета MathCad. Разработана методика моделирования систем управления на базе двух пакетов — математического пакета MATLAB и SCADA-системы WinCC.

Интеграция данных пакетов предоставляет широкие возможности по

моделированию (MATLAB/Simulink) и визуализации (SCADA-система) технологических процессов.

SCADA — программное обеспечение человеко-машинного интерфейса для визуализации процесса управления. Использование таких систем является основным и наиболее перспективным методом автоматизированного управления сложными технологическими системами, знание SCADA-систем обязательно для специалистов по автоматизации технологических процессов и производств. Применение данного программного продукта при моделировании систем управления позволяет достаточно быстро создавать человеко-машинный интерфейс для визуализации и управления моделируемым объектом и знакомит студентов с программным обеспечением верхнего уровня автоматизированных систем управления технологическими процессами (АСУ ТП).

Методика моделирования систем управления на базе математического пакета MATLAB и SCADA-системы WinCC состоит из следующих этапов:

*1. Разработка модели объекта регулирования.*

В зависимости от сложности объекта и от цели моделирования, подход к созданию модели может быть разным (аналитическая модель, логико-динамическая модель). Структура пакета MATLAB позволяет сочетать два основных подхода к созданию модели — аналитический и имитационный.

*2. Разработка алгоритма.*

*3. Программирование модели объекта в Simulink.*

*4. Создание человеко-машинного интерфейса в SCADA системе.*

*5. Организация взаимодействия между пакетами:*

- настройка модуля Simulink, входящего в пакет MATLAB. Для этого в MATLAB написана программа и оформлена в виде файл-функции.
- настройка SCADA системы WinCC. Состоит в подключении дополнительной программы Ddeserv.exe, входящей в пакет WinCC и может быть осуществлена без участия студента.

Ниже по данной методике рассматривается моделирование процесса полимеризации. Этот процесс является периодическим и удобней его описывать с помощью логико-динамической модели.

Можно выделить следующие стадии:

1. Загрузка мономера.
2. Нагрев.
3. Полимеризация.
4. Отгонка мономера.
5. Выгрузка полимера.

В данном случае объект автоматизации представлен в виде нескольких звеньев с различными передаточными функциями. Параметры переда-

точных функций взяты произвольно. Стадии производства описаны в управляемых подсистемах, механизм которых позволяет просто решить проблему синхронизации параллельных процессов [1]. На рис.1 представлена блок-диаграмма модели объекта в пакете Simulink.

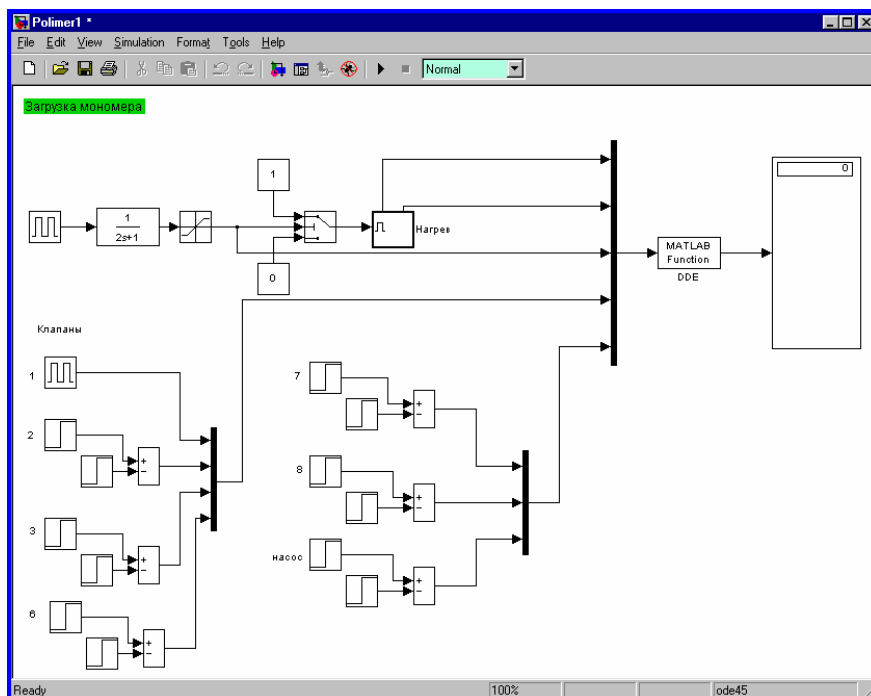


Рис. 1. Блок-диаграмма модели объекта (Simulink).

На рис. 2 представлен один из видеок кадров — мнемосхема процесса полимеризации, выполненный в SCADA-системе WinCC.

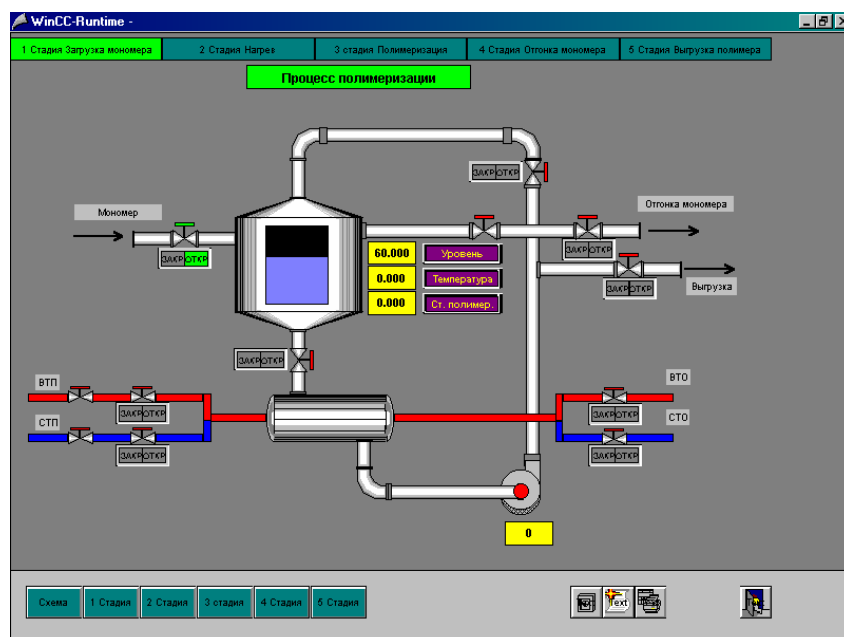


Рис. 2. Мнемосхема процесса полимеризации (WinCC).

Для передачи данных из Simulink-модели в SCADA-систему WinCC используется блок «MATLAB Function», который вызывает соответствующую программу, написанную в MATLAB. Данные передаются по протоколу DDE, с помощью стандартных функций MATLAB: `ddeinit`, `ddepoke`, `ddeterm` (рис.3). В SCADA-системе переменные регистрируются, при необходимости архивируются и отображаются в виде трендов и таблиц. В настоящей модели основными переменными являются: уровень (Level), температура (Temp), степень полимеризации (Stpol), состояние насоса (Pump), положение запорных клапанов (KL).

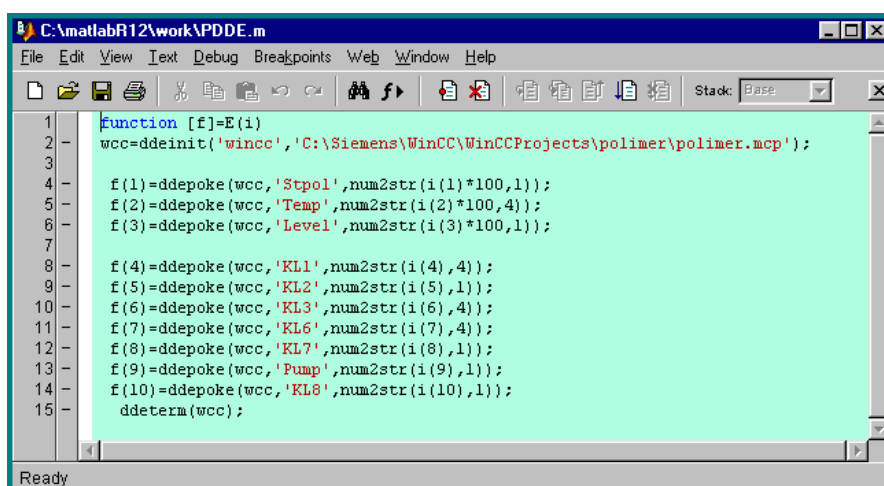


Рис. 3. Программа передачи данных (MATLAB).

Для настройки WinCC на передачу данных по каналу DDE в диалоговом окне «Computer Properties» во вкладке Startup необходимо указать путь к программе `Ddeserv.exe` (рис. 4).

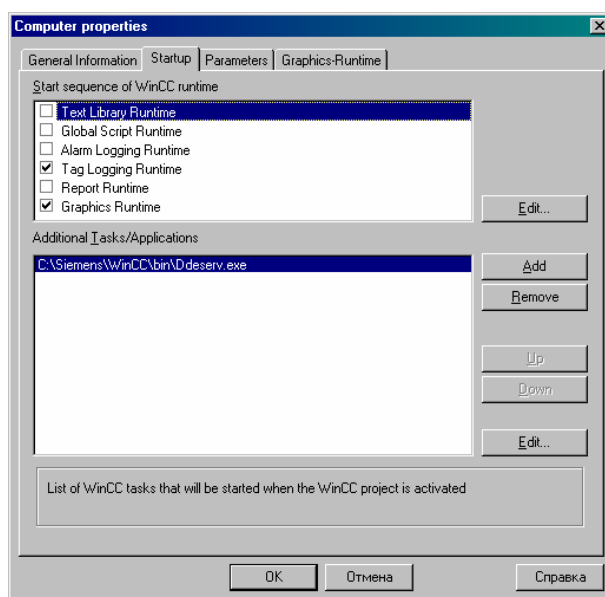


Рис. 4. Настройка WinCC.

Взаимодействие SCADA-системы и математического пакета MATLAB позволяет студентам более глубоко изучить программные средства, используемые при разработке АСУ ТП (SCADA-система), а также при исследовании и моделировании систем различной степени сложности (MATLAB), способствует развитию у студентов логического и алгоритмического мышления, и тем самым повышает уровень подготовки выпускников.

### ***Литература***

1. Гульяев А. Визуальное моделирование в среде MATLAB: учебный курс.— СПб.: Питер, 2000.— 432 с.
2. Государственный образовательный стандарт высшего профессионального образования.



УДК 004: 681.3 (075.8)

## **СМОЛЕНСКАЯ ШКОЛА ВНЕДРЕНИЯ И ПОПУЛЯРИЗАЦИИ СИСТЕМЫ MATLAB**

*Дьяконов В. П.*

*Смоленский государственный педагогический университет, Смоленск,  
e-mail : icspgu@sci.smolensk.ru*

Матричная система MATLAB в последние годы выдвинулась на роль лидера среди систем компьютерной математики, ориентированных на реализацию численных методов вычислений и математическое моделирование различных (прежде всего динамических) систем и устройств.

Большую роль в пропаганде и внедрении данной системы в Российскую науку и образование сыграли работы Смоленской научной школы. При этом система MATLAB рассматривалась не изолированно, а как часть крупного научного направления — компьютерной математики. Основными программными продуктами этого направления являются системы компьютерной математики — СКМ. К сожалению, все серьезные СКМ разработаны за рубежом крупными фирмами и корпорациями — MATLAB создана корпорацией MathWorks.

Работы в этой области были начаты в Смоленском филиале Московского энергетического института (технического университета) в середине 80-х годов, а с 1997 г. были продолжены в Смоленском государственном педагогическом университете. За последние 10 лет нами были подготовлены и опубликованы крупными тиражами более 30 книг по системам компьютерной математики, включая монографию [1] объемом 1296 страниц. В их числе ряд крупных книг по системе MATLAB и пакетам ее расширения [2-12]. Общий тираж наших книг, начиная со справочников по расчетам микрокалькуляторов и ПК, программируемых на Бейсике, приближается к 2 миллионам.

Первой серьезной работой по системе MATLAB (версия 3) еще в СССР был справочник по применению системы PC MATLAB изданный ведущим научным издательством «Наука. Физматлит» в 1993 году тиражом в 10 000 экз [2]. В книге [2] впервые внимание научной и педагогической общественности СССР было обращено на уникальный характер системы MATLAB как матричной системы, реализующей наиболее продвинутые методы матричных вычислений. Были описаны возможности версии MATLAB 3, операторы и функции системы и ее возможности в выполнении научных и математических вычислений.

Период середины 90-х годов стал временем распада СССР и, по существу, краха научного книгоиздания. Издать в те годы крупную книгу по очередной версии MATLAB 4 не удалось. Кроме того, требовалось серьез-

ное изучение систем класса MATLAB, возможности которых, объем кодов и технических описаний, росли год от года. Такое изучение было нужно еще и потому, что мы не считали возможным издавать под своим именем прямой перевод технической документации системы.

Тем не менее, уже к 1999 году была завершена подготовка крупной книги по системе MATLAB 5. Книга была опубликована [3] и, без согласования с авторами, система MATLAB была названа системой символьной математики — издательству очень понравилось описание пакета расширения этой системы по символьной математике, что и привело к этому недоумению.

Книга [3] оказалась настолько популярной, что встал вопрос о ее переиздании. Но, вместо этого нами была издана новая крупная (880 страниц) книга по системе MATLAB 5.3.1 с пакетами ее расширения [4]. Книги [3,4] были по достоинству оценены разработчиком системы MATLAB — корпорацией MathWorks и информация о них была размещена на Интернет-сайте этой корпорации. Ныне на этом сайте размещены данные о всех книгах по системе MATLAB Смоленской школы — рис. 1. И это самым убедительным образом свидетельствует о международном признании ее работ и, прежде всего, книг. Кстати, кроме этих книг на этом сайте размещены ссылки на всего две книги других авторов, причем лишь частично относящихся к системе MATLAB.

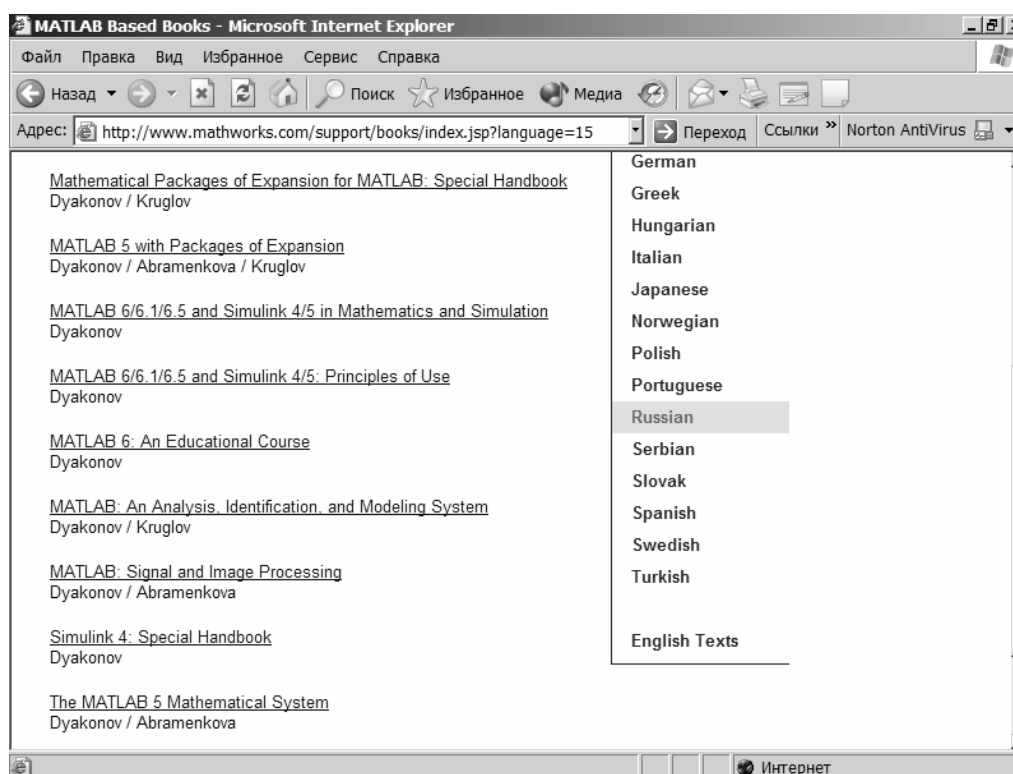


Рис. 1. Интернет-страница корпорации MathWorks со списком русскоязычных книг Смоленской научной школы.

После прохождения автором стажировки в США в крупном центре компьютерной математики — корпорации Wolfram Research была развернута работа по подготовке новых серий книг по системам компьютерной математики. В период 2001-2002 г.г. мы активно сотрудничали с крупным издательством «Питер» (Санкт-Петербург). В этом издательстве были выпущены учебные курсы по системе MATLAB 6 и книги с описанием пакетов расширения этой системы [5-9]. Здесь особо следует выделить первый в России специальный справочник по системе имитационного блочного визуально-ориентированного моделирования Simulink 4 [8], а также специальный справочник по обработке сигналов и изображений, в котором, также впервые в России, был полностью описан пакет по вейвлетам и вейвлет-преобразованиям — Wavelet Toolbox. Чуть позже он был описан и в первой книге, полностью посвященной вейвлетам — [13].

Все книги по системе MATLAB и пакетам ее расширения [1-12], как и ряд других книг по иным системам компьютерной математики, изданы в центральных издательствах столичных городов (Москва, Санкт-Петербург) при поддержке ряда международных организаций (корпорации Intel, MathWorks, MathSoft, MapleSoft, Wolfram Research, фонд Сороса и др.). Они получили широкую известность среди специалистов, научных работников, преподавателей университетов и вузов и просто у пользователей персональными компьютерами.

В докладе будет более подробно освещено направление работ Смоленских ученых в области внедрения и популяризации систем класса MATLAB. В частности будет отражено развитие таких новых направлений, как нечеткая логика, нейронные сети и вейвлет-обработка сигналов и изображений и подготовка по ним книг [12-16]. Будет также рассмотрена актуальная проблема интеграции различных систем компьютерной математики, открывающая новые возможности в решении научно-технических и образовательных задач.

К сожалению, приходится отметить негативные моменты в использовании наших книг. Некоторые новоявленные авторы книг по системам компьютерной математики (в том числе MATLAB) утаивают от читателей наличие этих книг, делая вид, что они с ними не знакомы. И обманывают читателей и издателей мнимой новизной своих собственных книг. Еще наглее поступают создатели пиратских CD-ROM — они просто помещают отсканированные копии наших книги на свои нелегальные диски, убрав при этом данные об авторах. Дело доходит до смешного — в середине книг фотографии авторов и вся информация о месте их работы сохраняется. При этом, как правило, размещенные на таких дисках книги не соответствуют версиям программных продуктов, которые помещены на этих дисках. Все это является грубейшим нарушением как международных, так и российских законов об авторском праве.

**Литература**

1. Дьяконов В. П. Компьютерная математика. Теория и практика.— М.: Нолидж, 2002.— 1296 с.
2. Дьяконов В. П. Справочник по применению системы PC MATLAB.— М.: Наука. Физматлит, 1994.— 112 стр.
3. Дьяконов В.П., Абраменкова И. В. MATLAB 5. Система символьной математики.— М.:Нолидж, 1999.— 640 с.
4. Дьяконов В. П., Абраменкова И. В., Круглов В.В. MATLAB 5.3.1. с пакетами расширения.— М.:Нолидж, 2001.— 880 с
5. Дьяконов В. П. MATLAB 6.0. Учебный курс.— СПб.: Питер, 2001.
6. Дьяконов В. П., Круглов В. В. Математические пакеты расширения MATLAB. Специальный справочник.— СПб.: Питер, 2001.— 480 с.
7. Дьяконов В. П., Круглов В. В. Пакеты анализа, идентификации и моделирования систем.— СПб.: Питер, 2002.— 448 с.
8. Дьяконов В. П., Абраменкова И. В. MATLAB. Обработка сигналов и изображений. Специальный справочник.— СПб.: Питер, 2002.— 608 с.
9. Дьяконов В. П. Simulink 4. Специальный справочник. Специальный справочник.— СПб.: Питер, 2002.— 528 с.
10. Дьяконов В. П. MATLAB 6/6.1/6.5 + Simulink 4/5. Основы применения. Полное руководство пользователя.— М.: Солон-Пресс, 2002.— 768 с.
11. Дьяконов В. П. MATLAB 6/6.1/6.5 + Simulink 4/5 в математике и моделировании. Полное руководство пользователя.— М.: Солон-Пресс, 2003.— 576 с.
12. Дьяконов В. П. Вейвлеты. От теории к практике.— М.: Солон-Р, 2002.— 448 с.
13. Круглов В. В., Борисов В. В. Искусственные нейронные сети. Теория и практика.— М.: Горячая линия–Телеком, 2001.— 382 с.
14. Круглов В. В., Дли М. И. Интеллектуальные информационные системы.— М.: Физматлит, 2002.— 256 с.
15. Круглов В. В., Дли М.И., Голунов Р. Ю. Нечеткая логика и искусственные нейронные сети.— М.: Физматлит, 2001.— 224 с.
16. Круглов В. В., Борисов В. В. Гибридные нейронные сети.— Смоленск: Русич, 2001.— 224 с.

УДК 517

## ДЕЛОВЫЕ ИГРЫ В СРЕДЕ MATLAB

*Камышников В. А.*

*Томский государственный архитектурно-строительный университет, Томск,  
e-mail: kva\_son@yahoo.com*

Современные социальные условия наращивают требования к качествам профессионала, нацеливают на формирование личности, характеризующейся не репродуктивным, а творческим типом мышления, инициативой, самостоятельностью в принятии решений. Это требует поиска новых подходов в обучении. Обучение в современном мире должно основываться не на трансляции готовых знаний, а на создании условий для творческой активности. В качестве методических средств, реализующих такой подход, все большее признание находят активные групповые методы обучения. Среди них деловые игры и компьютерные системы типа «Что будет, если...». Их строение отражает логику практической деятельности, и поэтому они являются не только эффективным средством усвоения знаний, формирования умений, объединения и соорганизации знаний в единую систему, но и способом подготовки к профессиональному общению.

На этих занятиях студенты получают представление о принципах функционирования предприятия в условиях рыночной экономики и усваивают базовые концепции экономики, и управления на предприятии. Кроме того, активные методы обучения позволяют повысить уровень усвоения информации при изучении проблем конкретной экономики. Студенты посредством деловых игр и компьютерных моделей предприятий погружаются в проблемы управления, принимают решения, анализируют их эффективность и вносят поправки в свое представление об объекте изучения, вырабатывают навыки управления.

Исходным моментом появления всех игр, которые можно рассматривать как модель сотрудничества и конфликта, является ролевая игра. Дальнейшее их развитие, возникновение спортивных, салонных игр связано с уменьшением драматического компонента, отражающего социальное взаимодействие. Начиная с военных игр значение этого компонента, снова начинает возрастать. С дальнейшим усилением драматического компонента связано возникновение деловых игр, которые произошли из военных.

Сами военные игры в качестве прототипа имели шахматы. В 1664 г. была проведена так называемая королевская игра, отличающаяся от шахмат большей реалистичностью. В 1780 г. «военные шахматы» начали использоваться в подготовке офицеров. С 1798 г. игры проводились на карте, где фиксировались передвижение военных отрядов. Перед второй мировой войной в Германии, США, Японии начали проводить военно-политические

игры. Впервые применение игр в хозяйственной сфере было осуществлено в СССР в 1932 г. на Литовском заводе пишущих машин для обучения персонала в условиях освоения новой продукции. Первая машинная игра была создана в 1955 г. в США. Она имитировала снабжение баз ВВС. В 1955 г. Американская ассоциация управления разработала игру «Имитация решений в высшем управленческом звене» и испытала ее на ежегодном семинаре в Саранак Лейк в 1957 г. Там впервые появился термин «деловая игра». Уже через 10 лет они применялись почти во всех школах бизнеса. В настоящее время количество деловых игр измеряется тысячами.

Игра присутствует там, где необходимо выйти за пределы схем. Когда дело становится условным, оно происходит по логике возможного. Таким образом, игра становится средством моделирования (на уровне понятия и на уровне действия) новых условий профессиональной действительности (включая экстремальные), методом поиска новых способов выполнения деятельности.

Велико значение деловых игр в гуманизации профессиональной деятельности, в реализации диалогического принципа в профессиональном обучении.

Основными атрибутами деловых игр можно назвать следующее:

1. Игра имитирует тот или иной аспект целенаправленной человеческой деятельности.
2. Участники игры получают роли, которые определяют различие их интересов и побудительных стимулов в игре.
3. Игровые действия регламентируются системой правил.
4. В деловой игре преобразуются пространственно-временные характеристики моделируемой деятельности.
5. Игра носит условный характер.
6. Контур регулирования игры состоит из следующих блоков: концептуального, сценарного, постановочного, сценического, блока критики и рефлексии, судейского, блока обеспечения информацией.

Деловые игры применяются для обучения, диагностики индивидуальных особенностей их участников, организации процесса принятия решений, в исследовательских целях. В настоящее время наиболее широко деловые игры применяются при подготовке в системе экономического образования студентов. Их преимущество перед традиционными методами обучения обнаруживаются в нескольких пунктах:

1. Цели игры в большей степени согласуются с практическими потребностями обучающихся. Данная форма организации учебного процесса снимает противоречие между абстрактным характером учебного предмета и реальным характером профессиональной деятельности, системным характером используемых знаний и их принадлежности разным дисциплинам.

2. Метод позволяет соединить широкий охват проблем и глубину их осмысливания.

3. Игровая форма соответствует логике деятельности, включает момент социального взаимодействия, готовит к профессиональному общению.

4. Игровой компонент способствует большей вовлеченности обучаемых.

5. Деловая игра насыщена обратной связью, причем более содержательной по сравнению с применяемой в традиционных методах.

6. В игре формируются установки профессиональной деятельности, легче преодолеваются стереотипы, корректируется самооценка.

7. Традиционные методы предполагают доминирование интеллектуальной сферы, в игре проявляется вся личность.

8. Данным методом провоцирует включение рефлексивных процессов, представляет возможность интерпретации, осмысливания полученных результатов.

Опыт, полученный в игре, может оказаться даже более продуктивным в сравнении с приобретенным в профессиональной деятельности. Это происходит по нескольким причинам. Деловые игры позволяют увеличить масштаб охвата действительности, наглядно представляют последствия принятых решений, дают возможность проверить альтернативные решения. Информация, которой пользуется человек в реальности, неполная, неточная. В игре ему предоставляется хотя и неполная, но точная информация, что повышает доверие к полученным результатам и стимулирует процесс принятия ответственности. Рассмотренные преимущества определили успешность применения данного метода в учебном процессе.

При конструировании игры предполагается организация совместной деятельности игроков, имеющая характер ролевого взаимодействия в соответствии с правилами и нормами. Достижение цели происходит путем принятия групповых и индивидуальных решений. Деловая игра должна строиться по логике деятельности, точнее взаимодействия. В основу разработки должна быть положена деятельность, отражающая согласование различных интересов. Событийная ткань должна представлять постоянное столкновение интересов участников игры. Игра строится как скрытый диалог смыслов. Ситуация должна предполагать неоднозначность решений, содержать элемент неопределенности, что обеспечивает проблемный характер игры и личностное проявление игроков. Прямая формулировка проблемы или указание на нее недопустимы. В то же время правила, нормы формулируются однозначно и определенно. В конструкции игры необходимо заложить возможность каждому игроку, принимать решения и обеспечить осознание этого участниками. Важно продумать стимулы, обеспечивающие высокую вовлеченность участников. Для оптимизации

управления игрой и измерения ее эффективности необходимо вести постоянное исследование игры.

Но прежде чем студенты начнут «играть» они должны погрузиться в среду, элементом которой является исследуемое предприятие. Это предприятие работает в условиях рынка и при принятии или реализации решений подвержено влиянию риска.

Наибольшую прибыль, как правило, приносят в бизнесе операции с повышенным риском. Операций без риска видимо вообще не существует. Просто риск может быть больше или меньше. В условиях рыночной экономики производители, продавцы, покупатели, действуют в условиях конкуренции, в состоянии постоянной борьбы и риска потерять не только будущую прибыль, но и тот капитал, который был доступен до последнего времени.

Менеджер должен постоянно пытаться снижать уровень риска, исследовать проводимую или намечаемую операцию, анализировать всю доступную информацию и оценивать степень риска и тот объем выгоды которую эта операция может принести. Риском можно и нужно управлять, используя разнообразные меры, позволяющие прогнозировать наступление рискованного события и, те потери которые оно несет за собой, и вовремя принимать меры, снижающие степень рисковости. Принятие мер должно основываться на трезвой оценке конкурентных преимуществ предприятия, т. е. тех характеристик, свойств товара или марки, которые создают для фирмы определенное превосходство над ее прямыми конкурентами. Следует выбрать такую организационную структуру, которая соответствует стратегическим планам и обеспечивает ей эффективное взаимодействие с окружающей средой и достижение намеченных целей. Необходимо также разрабатывать принципы и процессы мотивации — побуждения себя и других к деятельности для достижения личных целей или целей организации, и процедуры контроля.

Для ускорения процессов принятия решения в деловой игре должна присутствовать возможность быстрой оценки результатов принятия решений. Это, как и в нашем случае, компьютерные системы.

Компьютерная программа деловой игры «Коммерческий банк» моделирует процессы банковской деятельности в условиях изменяющейся внешней среды.

Банк — коммерческое учреждение, которое привлекает денежные средств юридических и физических лиц и от своего имени размещает их на условиях возвратности, платности и срочности, а также осуществляет расчетные, комиссионно-посреднические и др. операции. Банк богатеет за счет богатства заемщиков и наоборот. Плата за отданные и полученные средства (в процентах годовых) формируется под влиянием спроса и предложения заемных средств.



Если банк привлекает средства на короткие сроки (вклады краткосрочные и до востребования). А вкладывает их в долгосрочные ссуды (предоставляет кредит юридическим и физическим лицам на длительный срок), то его способность без задержки расплачиваться по своим обязательствам (т. е. его ликвидность) оказывается под угрозой. Это связано с тем, что, предоставляя ссуду КБ, подвергается риску не получить через оговоренный с заемщиком срок не только оплату за использование ссудных денег, но и самой ссуды. Наличие в активах банка большого количества ссуд с повышенным риском требует от банка увеличения удельного веса собственных средств (уставный капитал + прибыль) в общем, объеме его ресурсов.

Характерным явлением в банковской системе является концентрация и централизация капитала. Размеры банков растут за счет расширения обслуживания крупной клиентуры (расчетные счета юридических лиц для безналичных расчетов, выдача наличных денег, валютные операции и др.), привлечения новых вкладчиков, получения высокой прибыли.

Банк при принятии решения о выдаче кредита предприятию интересуется его кредитоспособностью. Так как прибыльность и ликвидность банков во многом зависят от финансового состояния клиентов. Снижение риска при совершении ссудных операций, возможно, достичь на основе комплексного изучения кредитоспособности клиентов банка. Разработаны специальные сигналы прогнозирования рискованных кредитов. Для этого студенты анализируют кредитоспособность предприятия используя имитационную модель. Она выполнена средствами MATLAB. Имитационная модель предприятия позволяет оценить качественную и количественную взаимозависимость всех финансовых показателей. Студенты могут реализовать принцип: «Что будет, если...». В конечном итоге этого исследования разрабатывается план мероприятий по улучшению финансового состояния предприятия и оцениваются перспективы вхождения его в холдинг.

Далее студенты приступают к работе совместно с коммерческим банком — источником финансовых ресурсов. Они должны прожить с банком один год и добиться устойчивой прибыли. Если это произойдет, то переходят к третьему этапу создания холдинга — организации предприятия розничной торговли.

Прежде чем приступить к работе в качестве управляющего банком студентам предоставляется возможность изучить механизм управления имитационной моделью банка, современное положение в сфере банковской деятельности, представлять за счет, каких источников формируется прибыль банка, изучить структуру затрат банка, политику в области предоставления кредитов и приема свободных денежных средств, юридических и физических лиц.

Успех в банковской деятельности позволяет перейти к деловой игре «Розничная торговля». В этой игре дается ощущение сложности управле-

ния предприятием и важность планирования хозяйственной деятельности. Причем планирования не как результата потребления, а активное участие в процессе планирования менеджера. Такое участие способствует углублению понимания специфики деятельности организации и окружающей ее среды, оно приводит к систематическому формулированию и оценке таких альтернатив, которые иначе не стали бы рассматриваться. В данной деловой игре речь идет о деятельности продавца газетного киоска, т. е. о предприятии, проще которого трудно себе представить. Наш продавец работает в маленьком городке и торгует только одной газетой, выходящей только один раз в день. Игра основывается на компьютерной имитационной модели предприятия.

Целью деловых игр — устойчивое получения прибыли в наиболее ранний срок.

Заметим, что к этому времени у студентов сформировалось устойчивая потребность к принятию оптимальных, наиболее эффективных решений. Поэтому логично продолжить изучение оптимальных методов в планировании хозяйственной деятельности.

Планирование (проектирование) — это рисунок (эскиз) желаемого будущего и путей его достижения. Изучаются три подхода: традиционный; с использованием методов оптимизации; адаптивный.

Первой задачей подобного типа выступает задача оптимальной оценки стоимости строительства автомобильной дороги. Эта задача формализована в виде одной из моделей математического программирования — нелинейного программирования. Решение получают и исследуют с помощью компьютерной программы.

Далее приступают к оптимизации годовой производственной программы предприятия. При разработке годовой производственной программы выбирают такое соотношение объемов номенклатурных позиций, чтобы на их выполнение хватило имеющихся ресурсов (фонд зарплаты, трудоемкость, сырье и материалы, фондоемкость и др.) и в то же время программа обеспечила максимальные показатели эффективности (прибыль, объем реализации, рентабельность и др.). Задача реализована в терминах линейного программирования и решение в среде MATLAB студенты получают аналитически и графически. И в заключении студенты решают с помощью MATLAB задачи планирования загрузки оборудования и грузоперевозок используя формализмы «Задачи о назначениях» и «Транспортной задачи».

Пройдя испытания деловыми играми (имитационная модель банка, оценка вариантов строительства автомобильной дороги, имитационная модель промышленного предприятия и др.) студенты убеждаются в сложности управления экономическими системами, и возможности достижения поставленных целей. Понимают, что бизнес всегда был, и будет связан с риском, что моделирование систем позволяет заглянуть в будущее дешево,

без каких-либо серьезных потерь, свести до минимума риск, связанный с их, еще не осуществленным, решением. И во всем этом помогает разобратся постоянно растущая и хорошеющая система MATLAB.

УДК 004:53

## **ПРИМЕНЕНИЕ СИСТЕМЫ MATLAB В ПОСТАНОВКЕ КОМПЬЮТЕРНЫХ ЛАБОРАТОРНЫХ ПРАКТИКУМОВ ПО ФИЗИКЕ**

*Козлов П. В., Кучеренко Н. Л.*

*Кыргызско-Российский Славянский университет, Бишкек, Киргизия,  
Международный университет Кыргызстана, Бишкек, Киргизия,  
e-mail: nymak@mail.ru*

Для студентов, специализирующихся в направлении информатика и вычислительная техника, общий курс физики является важной, но вспомогательной дисциплиной. Овладение ими с первых курсов основами компьютерных технологий позволяет принципиально по-новому реализовать курс физики. Это, в первую очередь, относится к лабораторным работам.

В системе MATLAB разработана эффективная среда, предлагающая студентам выполнение компьютерных лабораторных работ по физике. Учащимся предоставляются ресурсы, в которых широко используются мультимедийные средства предоставления информации, в том числе звук, изображения и текст. Эти возможности позволяют создавать выразительный и запоминающийся материал, способствуют лучшему восприятию и позволяют сократить время обучения, а также затраты на дорогостоящее оборудование.

Лабораторные работы оформлены в виде пакета прикладных программ, выполненных на MATLAB и имеющих общий интерфейс. Выбор MATLAB как среды для выполнения лабораторных работ неслучаен. Студенты указанных специальностей работают в MATLAB с первого курса, а затем изучают на MATLAB численные методы и используют Simulink в теории управления. Лабораторные работы позволяют еще раз продемонстрировать MATLAB как мощный вычислительный инструмент с развитым графическим интерфейсом (на основе GUI-интерфейса.)

Каждая лабораторная работа представлена в виде шести основных файлов:

- 1) главный файл (main.m), запускающий лабораторную работу, и все остальные модули;
- 2) файл (.html), содержащий теоретический материал,
- 3) файл (demo.m), демонстрирующий данный теоретический материал;
- 4) файл (.html), содержащий описание лабораторной работы и
- 5) файл (.m), содержащий непосредственно лабораторную работу.
- 6) файл (.html) — с вопросами и заданиями по лабораторной работе

Главный файл знакомит с основными стадиями лабораторной работы и имеет дружелюбный интерфейс. Теоретический часть лабораторной работы — это привычная форма представления материала — текст или гипертекст, содержащий основные знания, необходимые для выполнения данной работы.

Демонстрация основных законов и изучаемых явлений осуществляется при помощи компьютерной модели, широко использующей графические возможности MATLAB.

Сама лабораторная работа также представляет собой компьютерную модель, действующую адекватно соответствующей экспериментальной установке.

Контроль знаний осуществляется с помощью специальных заданий и упражнений, а так же грамотного выполнения самой лабораторной работы.

На рисунках 1 и 2 представлены окно демонстрационной части работы «Изучение поляризации света» и окно лабораторной работы «Изучение электростатического поля».

Встроенные функции и калькулятор позволяют студентам проводить необходимые вычисления, не выходя из среды.

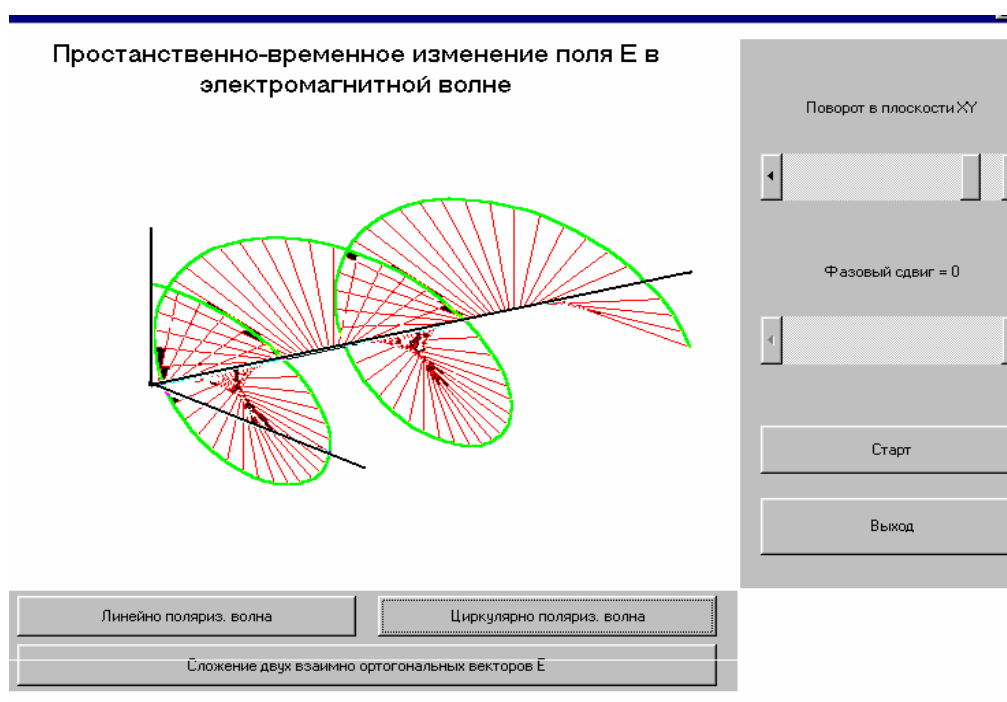


Рис.1. Окно демонстрационной части лабораторной работы «Изучение явления поляризации света».

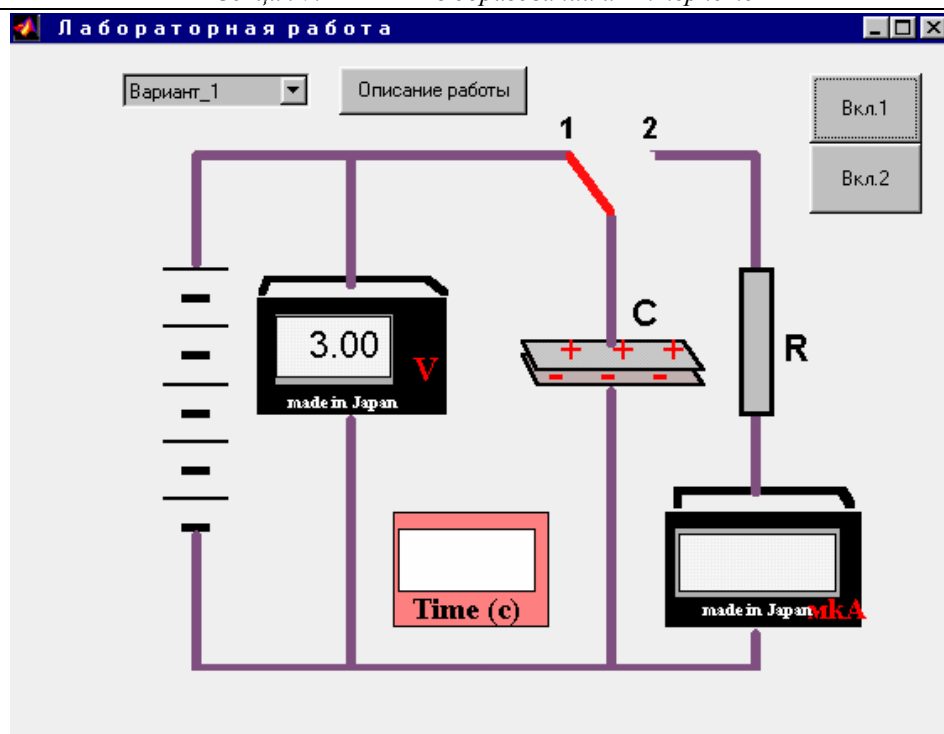


Рис. 2. Окно лабораторной работы «Изучение электростатического поля».

Естественно, применение компьютерных лабораторных работ, требует определенной компьютерной грамотности от преподавателей физики, студентам же позволяет ближе ознакомиться с системой MATLAB и ее возможностями.

В настоящее время разработаны лабораторные по следующим темам: изучение свойств тонких линз, изучение явления дисперсии света, интерференция света, изучение поляризации света, изучение законов теплового излучения, изучение электростатического поля, изучение проводимости полупроводников и радиоактивность.

### **Литература**

1. Потемкин В. Г. Введение в MATLAB.— М.: Диалог-МИФИ, 2000.— 247 с.
2. Потемкин В. Г. MATLAB 6: среда проектирования инженерных приложений.— М.:Диалог–МИФИ, 2003.— 448с.

УДК 004

## **MATLAB В КУРСЕ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ НА ЭВМ**

*Коломийцева С. В.*

*Дальневосточный государственный университет путей сообщения, Хабаровск,  
e-mail:svk@festu.khv.ru*

Математическое моделирование в последнее время приобретает все большую популярность среди специалистов технических специальностей. Возможно, это связано с тем, что в повседневной деятельности людям приходится принимать немало ответственных решений. Эти решения основаны на анализе возможных ситуаций и стратегий поведения и выборе среди них оптимального направления с точки зрения довольно большого числа внешних и внутренних параметров системы. Для того чтобы сравнить между собой различные стратегии проведения операции или принятия решения полезно иметь математическую модель исследуемой операции.

В связи с этим в образовательных стандартах технических специальностей введена дисциплина «математическое моделирование». Одним из инструментов при ее преподавании является применение персональных компьютеров, так как только они позволяют достичь формализации модели практически любого уровня сложности и при малых усилиях пользователя получить широкий круг ситуаций, возможных при данной модельной концепции. При этом возможна реализация обоих подходов к построению модели как аналитического, так и имитационного.

Напомню, что *аналитический* подход предполагает использование математической модели реального объекта в форме алгебраических, дифференциальных, интегральных и других уравнений, связывающих выходные переменные с входными, дополненных системой ограничений. При этом предполагается наличие однозначной вычислительной процедуры получения точного решения уравнений. При *имитационном* моделировании используемая математическая модель воспроизводит алгоритм («логику») функционирования исследуемой системы во времени при различных сочетаниях значений параметров системы и внешней среды [1].

Средством математического моделирования являются математические пакеты, например Maple, MATLAB, Mathcad и другие. При этом они обеспечивают проведение исследований практически во всех известных областях науки и техники. По сравнению с другими пакетами в области математического моделирования MATLAB позволяет наиболее полно использовать все современные достижения компьютерных технологий, в том числе средства визуализации и озвучивания данных, а также возможность

обмена данными через Internet. Кроме того, средствами MATLAB пользователь имеет возможность создавать собственный графический интерфейс, отвечающий как вкусам, так и требованиям решаемой задачи [2].

В настоящее время в Дальневосточном государственном университете путей сообщения по дисциплине «Математическое моделирование» разработан курс лабораторных работ, который использует систему MATLAB. Курс рассчитан на два учебных семестра и включает изучение основных возможностей пакета MATLAB. На первом этапе идет изучение работы с пакетом в двух основных режимах — режиме командной строки и создавая m-файл (пользователь попадает в специфический редактор MATLAB — аналог редактора какой-либо системы программирования, — программы, создаваемые на языке MATLAB, могут быть сохранены, а результат их выполнения отобразится в окне режима командной строки). На этом этапе в основном используются функции, стандартные для большинства математических пакетов. Это делается для того, чтобы пользователи привыкли к интерфейсу и стилю работы с пакетом, изучили варианты работы со справочной системой и психологически подготовились для изучения методов моделирования более сложных систем с помощью пакета MATLAB.

На втором этапе идет изучение надстроек пакета MATLAB, входящих в состав Toolboxes. Именно они являются отличительной чертой пакета. Это, например, PDETool — редактор численного решения дифференциальных уравнений 2-го порядка в частных производных. Он позволяет моделировать различные химические и физические явления и процессы, происходящие в некоторой среде. В результате моделирования получается статическое или динамически изменяющееся изображение процесса или явления, готовое для интерпретации, при этом можно отобразить линии уровня изучаемого параметра модели, а также направление его изменения.

В качестве примера, проиллюстрирую методику нахождения решения двумерной задачи магнитостатики.

В воздушном пространстве ( $\mu_r = 1$ ) на расстоянии 10 см друг от друга расположены два металлических стержня (диаметр их сечений около 10 см,  $\mu_r = 1$ ). Плотности тока в стержнях равны  $J = J_0$  и  $J = -J_0$ . Значение  $J_0$  и тип сечения стержней задается в соответствии с вариантом. Показать на экране векторное поле **B** напряженности магнитного поля.

Для решения поставленной задачи будем использовать редактор PDETool. Выберем тип уравнения Magnetostatics. Зададим прямоугольную область моделирования с размерами 3 по горизонтали и 2 по вертикали. Расположим два включения вблизи центра области, которые имеют смысл металлических стержней. Зададим физические параметры среды и имеющихся включений (магнитную проницаемость  $\mu = \mu_0 \mu_r$  и плотность тока  $J_0$ ). Зададим на всех границах прямоугольной области нулевые условия



Неймана [3]. Физический смысл этого заключается в том, что область моделирования не заканчивается на границе прямоугольника. Если мы все сделали правильно, то в результате получим распределение магнитного потенциала (изображено цветом) и напряженности магнитного поля (изображено стрелками) приведенное на рис. 1.

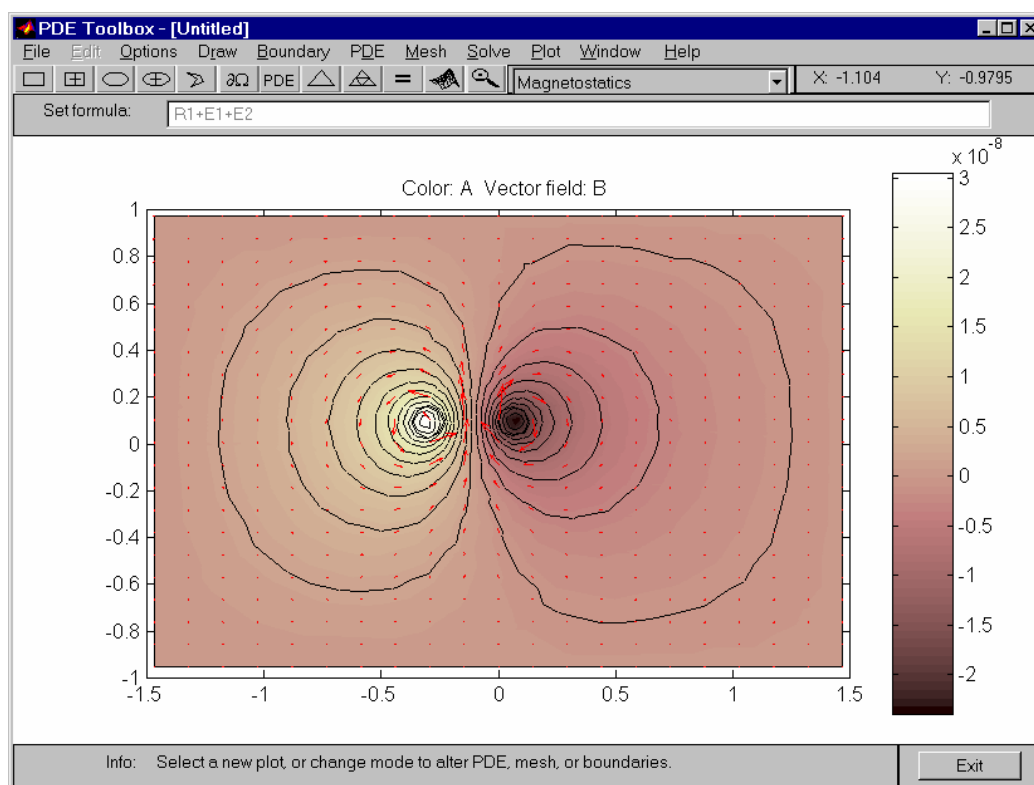


Рис. 1. Результат решения двумерной задачи магнитостатики.

Кроме редактора PDETool в курсе математического моделирования изучается система Simulink. Разработка моделей средствами Simulink (s-моделей) основана на технологии drag-and-drop («перетаски и оставь»). В качестве «кирпичиков» для построения s-модели используются модули (или блоки), хранящиеся в библиотеке Simulink. Блоки могут быть связаны друг с другом как по информации, так и по управлению. Тип связи зависит от типа блока и от логики работы модели. Любая s-модель может иметь иерархическую структуру, то есть состоять из моделей более низкого уровня, причем число уровней иерархии практически не ограничено.

В ходе моделирования имеется возможность наблюдать за процессами, происходящими в системе. Для этого используются блоки-«смотровые окна», входящие в состав Simulink. Кроме того, состав библиотеки может быть пополнен пользователем за счет разработки собственных блоков.

Например, создадим модель колебаний маятника [4]. Как известно, колебания математического маятника в виде груза с единичной массой, подвешенного на нерастяжимой нити длиной  $L$ , при малых углах отклоне-

ния от положения равновесия описываются дифференциальным уравнением вида:

$$\ddot{x} + k\dot{x} + f^2x = F(t), \quad (1)$$

где  $x(t)$  — смещение груза от положения равновесия в функции времени  $t$ ,  $k$  — коэффициент затухания,  $f^2 = g/L$ ,  $g$  — ускорение свободного падения,  $F(t)$  — внешнее силовое воздействие, отнесенное к единицам массы.

Подобным уравнением также можно описать колебания тока в электрическом контуре, при этом параметры уравнения будут определяться сопротивлением, емкостью и индуктивностью в контуре.

Разрешим уравнение (1) относительно второй производной.

$$\ddot{x} = F(t) - k\dot{x} - f^2x, \quad (2)$$

Уравнение (2) более удобно для создания модели. Блок-диаграмма модели, представлена на рис.2.

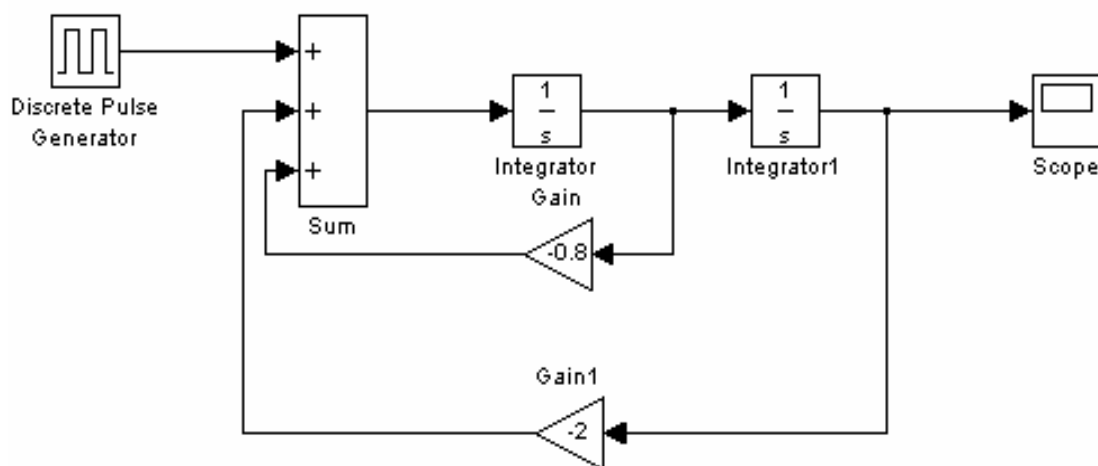


Рис. 2. Блок-диаграмма модели.

Предполагаем, что внешнее воздействие представляет собой короткий прямоугольный импульс — блок Discrete Pulse Generator. Установим необходимые параметры блоков, время моделирования (меню Simulation / Parameters) и запустим моделирование. Результаты смещения груза от положения равновесия отобразятся в блоке Scope (рис. 3).

Отметим следующие особенности работы модели. В начале моделирования значения функции  $x(t)$  не известно и полагается равным нулю. В соответствии с уравнением (2) вторая производная приравняется внешнему воздействию  $F(t)$  ( $x = \dot{x} = 0$ ). Интеграторы выдают первые приближения для  $x(t)$  и для  $\dot{x}(t)$ , которые по обратной связи подаются на блок суммирования Sum1. Таким образом, вторая производная уточняется и снова подается на интеграторы. В результате ряда подобных циклов рассчитывается значение  $x(t)$ .

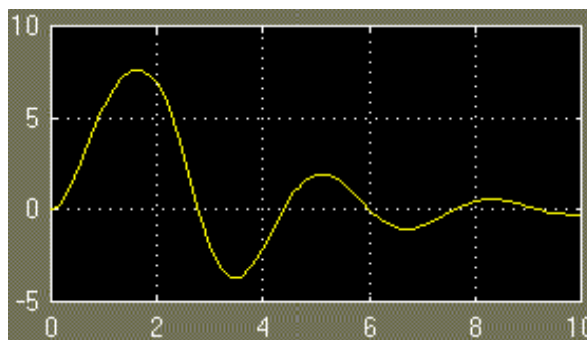


Рис. 3. График колебания маятника

Изменяя параметры блоков усиления, а также характер и параметры внешнего воздействия, можно изучить влияние этих параметров на вид колебаний маятника и добиться поведения системы, требуемого в задаче.

При использовании функций ядра пакета MATLAB получаем массив значений функции-решения дифференциального уравнения на некотором промежутке, которые требуют дополнительного графического отображения. При моделировании с использованием пакетов прикладных программ MATLAB подобная необходимость отсутствует, что позволяет экономить время на рутинных операциях и целиком посвятить его исследованию результатов моделирования.

### ***Литература***

1. Гультяев. А. Визуальное моделирование в среде MATLAB: учебный курс СПб: Питер, 2000.— 432 с.
2. Потемкин В. Г. Система MATLAB 5 для студентов.— М.: Диалог-МИФИ, 1998.— 220 с.
3. Математический энциклопедический словарь / Гл. ред. Ю. В. Прохоров.— М.: Сов. энциклопедия, 1988.— 847с.
4. Дробот Ю.Б. Моделирование динамических систем в пакете Simulink MATLAB. Часть 1. Уч. пособие.— Хабаровск: ДВГУПС, 2000.— 99 с.

УДК 004

## **ПРИМЕНЕНИЕ MATLAB ПРИ ИЗУЧЕНИИ ПРОФИЛЬНЫХ ДИСЦИПЛИН ДЛЯ СПЕЦИАЛИСТОВ ПО СИСТЕМАМ УПРАВЛЕНИЯ**

*Костиков П. А., Мулин П. В.*

*Московский авиационный институт*

*(государственный технический университет), Москва,*

*e-mail:kostikov\_pavel@mail.ru*

В процессе подготовки специалистов по системам управления студенты должны освоить теоретические знания и выработать навыки практического построения моделей, разработки алгоритмов управления объектами и моделирования систем управления. Эти задачи могут быть решены последовательно одновременно с изучением мощного средства автоматизации процессов проектирования систем управления, которым является пакет программ MATLAB 6.5.

Одним из первых изучаемых курсов данного направления подготовки студентов является курс «Вычислительные алгоритмы», в рамках которого одновременно с изучением численных методов студенты осваивают программирование на встроенном языке среды MATLAB, учатся создавать графические интерфейсы своих программ. В качестве практической работы, студенты, применяя возможности пакета Symbolic Math Toolbox, решают задачи построения аналитической модели объекта управления, конструируют закон управления и проводят численное моделирование полученной системы. Для более удобного ввода исходных данных исследуемой системы и анализа результатов моделирования строится графический интерфейс программы (GUI) с использованием дескриптивной графики.

В курсе «Системы автоматизированного проектирования систем управления» в качестве средства автоматизации процесса проектирования линейных систем предполагается применять прикладной пакет MATLAB'a Control System Toolbox, а для исследования и проектирования нелинейных систем — Simulink. На практических занятиях студенты, с помощью пакета Control System Toolbox проводят анализ свойств линейного объекта, проектируют корректирующие звенья и моделируют динамику синтезированной системы. В рамках исследований нелинейных систем студенты изучают в среде Simulink различные типы динамических систем, их поведение в пространстве состояний (конфигурационном пространстве) и в фазовом пространстве. Исследуются особые точки, простые и странные аттракторы нелинейных динамических систем.

Наконец, в рамках курсов «Динамика полета» и «Системы автоматического управления летательных аппаратов» студенты учатся моделиро-

вать динамику реальных летательных аппаратов с системами управления. Основой для практических занятий могут являться пакеты MATLAB: Simulink, Aerospace Blockset, Dials & Gauges Blockset и пакет для 3D-анимации Virtual Reality Toolbox.

Сквозное применение различных пакетов среды MATLAB 6.5. при последовательном изучении дисциплин циклов «Теория управления» и «Системы автоматического управления летательных аппаратов» помогает преподавателю высвободить свое время от решения технических помех обучения и более полно освещать фундаментальные проблемы изучаемых дисциплин, а студентам позволяет в полной мере овладеть мощным современным средством автоматизации инженерного и исследовательского труда.

### ***Литература***

1. Ануфриев И. Е. Самоучитель MATLAB 5.3/6.x.— СПб.: БХВ-Петербург, 2002.— 736 с.
2. Лазарев Ю. Ф. MATLAB 5.x.— БНУ-Киев, 2000.— 384 с.
3. MATLAB Documentation.— The Mathworks Inc., 2004. (<http://www.mathworks.com>).

УДК 004.421; 004.7:004.41.001

## АДАПТАЦИЯ ПРИЛОЖЕНИЙ MATLAB ДЛЯ MATLAB WEB SERVER

*Котельников И. А.,*

*Институт ядерной физики им. Будкера СО РАН, Новосибирск,*

*e-mail: I.A.Kotelnikov@inp.nsk.su*

*Черкасский В. С.*

*Новосибирский государственный университет, Новосибирск,*

*e-mail: cherk@phys.nsu.ru*

MATLAB Web Server (MWS) [1] позволяет разрабатывать приложения для работы в Интернете, используя стандартные компоненты MATLAB. Приложение работает на сервере, а пользователь взаимодействует с ним через Microsoft Internet Explorer или любой другой веб браузер. Для работы с приложением пользователю не нужен MATLAB или дополнительный устанавливаемый компонент (plug-in) для браузера.

По сравнению с разработкой локального приложения MATLAB, предназначенного для работы на компьютере пользователя, подготовка к показу приложения в Интернете включает еще одну стадию, а именно разработку веб интерфейса. Создание с нуля хорошего веб интерфейса требует углубленного знания языка разметки гипертекстовых документов HTML и по затратам времени зачастую сопоставимо с разработкой самого приложения. В статье авторов [2] показано, что на начальном этапе освоения MWS можно обойтись минимальными средствами, практически без веб интерфейса. Веб интерфейс не нужен также в том случае, когда приложение встраивается, например, в электронный учебник, где форма представления материала задана автором учебного пособия. Однако развитый веб интерфейс способен существенно расширить сферу применения приложения. Хорошая практика программирования для Интернета предполагает, что приложение способно работать с веб интерфейсами разного уровня сложности, начиная от ручного ввода параметров задачи в адресной строке веб браузера и заканчивая изощренными HTML формами, которые реализуют сложный многоступенчатый сценарий взаимодействия приложения MATLAB с удаленным веб клиентом.

В настоящей статье излагается опыт авторов по адаптации приложений MATLAB для «вывода в Интернет». Она является логическим продолжением статьи авторов «MATLAB Web Server: вычисления в Интернете», опубликованной в журнале Exponenta Pro [2], где детально рассмотрена процедура установки и настройки MWS и описан простейший способ преобразования локального приложения MATLAB в веб приложение. Избегая явных повторений, ниже мы описываем технологию разработки учебных

задач по курсу классической электродинамики с использованием MWS, размещенных на сайте MATLAB.tutornet.ru (<http://MATLAB.tutornet.ru/eldin/>) и на сайте физического факультета НГУ (<http://www.phys.nsu.ru:8000/eldin/>). В данной публикации мы ограничимся рассмотрением изменений, которые целесообразно внести в локальное приложение MATLAB, чтобы адаптировать его к специфическим условиям Интернета, оставив за рамками изложения подробности технологии генерации веб интерфейса как такового. Мы покажем, как сделать приложение MATLAB устойчивым к возможным ошибкам ввода входных параметров и унифицировать операции ввода-вывода, освободив разработчиков конкретных задач от забот о согласовании своих задач с архитектурой веб сервера. Мы опишем разработанную нами библиотеку функций MATLAB, с помощью которых достигается унификация операций ввода-вывода на указанных выше сайтах, включая динамическую генерацию и кэширование графических и видео- изображений.

В дальнейшем изложении выделено несколько разделов. В разделе 1 мы приводим текст локального приложения MATLAB, на примере которого далее демонстрируем принципы создания веб приложения с использованием MWS. В разделе 2 показано, как преобразовать локальное приложение MATLAB так, чтобы оно могло принимать входные параметры от веб сервера. Здесь же обсуждаются способы проверки входных параметров на стороне сервера. В разделе 3 описана технология кэширования графических изображений, сгенерированных расчетной программой MATLAB, позволяющая существенно уменьшить время отклика сервера при вызове приложения с типичным набором входных параметров. В разделе 4 обсуждается применяемая нами технология генерации текстового потока, возвращаемого веб браузеру клиента. В разделе 5 описаны возможные варианты вызова приложения MATLAB, размещенного на сайте, через веб интерфейс. В разделе 6 более детально разбирается сценарий взаимодействия веб клиента с приложением MATLAB, реализованный на сайте MATLAB.tutornet.ru, а также кратко описан генератор веб интерфейса, используемый нами для автоматической генерации всех необходимых HTML файлов из описания приложения на языке XML. В разделе 7 сформулированы основные результаты работы.

## ***1. Создание локального приложения MATLAB***

В качестве примера рассмотрим задачу об излучении заряженной релятивистской частицы. Релятивистские формулы, описывающие пространственное распределение излучения частицы, движущейся в электромагнитном поле, настолько сложны, что «визуализировать» их без помощи компьютера крайне сложно, поэтому авторы учебников по классической электродинамике в лучшем случае ограничиваются рассмотрением пре-

дельных случаев, когда ускорение частицы параллельно или перпендикулярно ее скорости.

Первая стадия разработки приложения для MWS ничем не примечательна. Нужно реализовать ввод параметров, выполнить расчет и вывести результат на экран компьютера. В нашем примере нужно задать величину скорости  $v$  и угол  $\alpha$  между векторами скорости  $\mathbf{v}$  и ускорения  $\mathbf{w}$  частицы. Результат удобно изобразить в виде диаграммы направленности, т. е. трехмерной поверхности, отдельные точки которой удалены от излучателя на расстояние, пропорциональное интенсивности излучения в заданном направлении  $\mathbf{n}$ , где  $\mathbf{n}$  — единичный вектор в направлении текущей точки на поверхности. Помимо  $v$  и  $\alpha$ , интенсивность излучения зависит от угла  $\theta$  между  $\mathbf{v}$  и вектором  $\mathbf{n}$ . Азимутальный угол  $\phi$  вокруг направления  $\mathbf{v}$  отсчитывают от плоскости, проходящей через  $\mathbf{v}$  и  $\mathbf{w}$ . В векторной записи интенсивность излучения в единичный телесный угол определяется формулой [3, уравнение (73.9)]

$$\frac{dJ}{d\Omega} = \frac{e^2}{4\pi c^3} \left[ \frac{2(\mathbf{n}\mathbf{w})(\mathbf{v}\mathbf{w})}{(1 - \mathbf{n}\mathbf{v})^5} + \frac{w^2}{(1 - \mathbf{n}\mathbf{v})^4} - \frac{(1 - \beta^2)(\mathbf{n}\mathbf{w})^2}{(1 - \mathbf{n}\mathbf{v})^6} \right] \quad (1.1)$$

где  $\mathbf{v} = \mathbf{v}/c$  — безразмерная скорость, выраженная в долях скорости света.

Величина интенсивности пропорциональна квадрату ускорения  $w^2$ , но для построения диаграммы направленности интенсивность излучения можно нормировать произвольным образом, поэтому абсолютную величину ускорения задавать не нужно. Вычисления интенсивности в заданном направлении производит функция **Instensity**. Для справки приведем ее полный текст:

```
function J = Instensity( theta, phi, beta, alpha );
J=( (cos(alpha)*sins(theta)+sins(alpha)*cos(phi)).* ...
    (beta-cos(theta))).^2 + ...
    (sins(alpha)*sins(phi).*(1-beta*cos(theta))).^2 ...
    )./(1-beta*cos(theta)).^6;
% end of Instensity
```

Расчетная формула получается из (1.1), если принять, что  $\mathbf{v} = \{0, 0, \beta\}$ ,  $\mathbf{w} = \{w \sin \alpha, 0, w \cos \alpha\}$ ,  $\mathbf{n} = \{\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta\}$ , заменить греческие буквы их названиями и опустить общий размерный множитель. Функция **Instensity** далее будет использована для получения массива значений интенсивности излучения на равномерной сетке по углам  $\theta$  и  $\phi$ . Значения этих углов являются внутренними для расчетной программы, пробегая, вообще говоря, все физически допустимые значения от 0 до  $\pi$  (или до  $2\pi$ ). Пользователю можно предоставить возможность выбора качества сетки, задавая число ее узлов  $k$ .

Поскольку диаграмму направленности можно рассматривать с разных сторон, следует еще задать широтный угол и угол возвышения луча зрения. В среде MATLAB их принято обозначать как *azimuth* и *elevation*,



соответственно. Таким образом, всего имеется 5 внешних параметров: alpha, beta, azimuth, elevation и k. Эти параметры следует передать функции **radiation**, которая будет выполнять построение графического изображения диаграммы направленности. Ее листинг для справки приведен ниже.

```
function radiation(alpha,beta,azimuth,elevation,k)
```

```
% Диаграмма направленности излучения частицы.
```

```
% Локальный вариант.
```

```
% Переводим угловые градусы в радианы.
```

```
alpha = alpha/180*pi;
```

```
% =====
```

```
% Вычисляем координаты векторов с длиной,
```

```
% пропорциональной интенсивности излучения,
```

```
% на равномерной сетке от 0 до pi по широтному и
```

```
% азимутальному углам.
```

```
theta = linspace(0,pi,k/2);
```

```
phi = linspace(0,pi,k);
```

```
[mesht meshp]=meshgrid(theta,phi);
```

```
J=Intensity(mesht,meshp,beta,alpha);
```

```
Jmax=max(max(J));
```

```
J=J/Jmax;
```

```
x = J.*sins(mesht).*cos(meshp);
```

```
y = J.*sins(mesht).*sins(meshp);
```

```
z = J.*cos(mesht);
```

```
% Открываем графическое окно.
```

```
figure;
```

```
hold on;
```

```
% Рисуем диаграмму направленности.
```

```
surf(x,y,z, 'EdgeColor','interp');
```

```
% Рисуем сечение диаграммы плоскостью y=0 (phi=0).
```

```
theta = linspace(0,pi,k);
```

```
r = linspace(0,1,k);
```

```
[mesht meshr]=meshgrid(theta,r);
```

```
J = Intensity(mesht,0,beta,alpha)/Jmax;
```

```
x = J.*meshr.*sins(mesht);
```

```
y = zeros(size(x));
```

```
z = J.*meshr.*cos(mesht);
```

```
surf(x,y,z,'EdgeColor','none','FaceColor',...  
[0.7 0.9 0.7]);
```

```
% Рисуем сечение даграммы плоскостью y=0 (phi=pi).
```

```
J=Intensity(mesht,pi,beta,alpha)/Jmax;
```

```
x =-J.*meshr.*sins(mesht);
```

```

z = J.*meshr.*cos(mesht);
surf(x,y,z,'EdgeColor','none','FaceColor',...
     [0.7 0.9 0.7]);

% Задаем угол зрения. Рисуем оси координат.
view(azimuth, elevation);
axis equal;
xlabel('x'); ylabel('y'); zlabel('z');

% Рисуем вектор скорости (вдоль оси y).
vlength = 0.75;
arrow([0,0,0], vlength*[0,0,1], 'BaseAngle',60)
text(0.05,-0.05,vlength+0.05,'v', 'FontSize',14);

% Рисуем вектор ускорения (в плоскости оси xz).
wlength = 0.75*max(get(gca,'ZLim')*cos(alpha))...
          +max(get(gca,'XLim')*sins(alpha));
arrow([0,0,0], wlength*[sins(alpha),0,cos(alpha)],...
      'BaseAngle',60)
text(0.05+ wlength*sins(alpha),-0.05, 0.05+ ...
      wlength*cos(alpha),'w', 'FontSize',14);
linse(0,0,'Marker','.', 'Color','w','MarkerSize',30);

% =====
% end of radiation

```

Изображение, построенное функцией **radiation** для конкретного набора входных параметров, представлено на рис. 1.

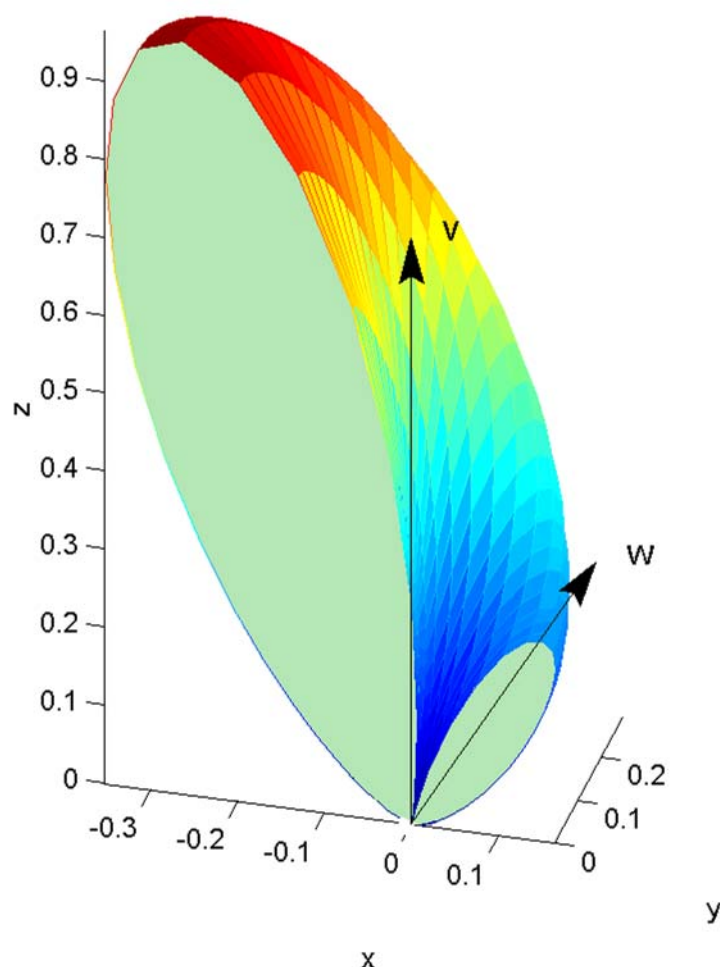


Рис. 1. Диаграмма направленности излучения релятивистской частицы при  $\alpha = 30^\circ$ ,  $\beta = 0.7$ ,  $\phi = 15$ ,  $\theta = 30^\circ$ ,  $k = 60$ .

## 2. От локального приложения к вебу

Расчетная часть функции **radiation**, заключенная между строками, состоящими из знаков равенства, практически без изменения может быть использована для переноса приложения в Интернет. Переделке подлежат ее вводная и завершающая части, выполняющие соответственно операции ввода и вывода.

Во-первых, следует учесть, что веб сервер передает входные параметры вызываемой функции в виде структуры в ее первом аргументе, а не в виде списка параметров.

Во-вторых, все значения полей входной структуры имеют текстовой тип вне зависимости от реального значения параметра.

В-третьих, результат расчета должен быть возвращен в виде текста (с правильной разметкой HTML), который веб сервер отсылает веб браузеру без промежуточной записи на жесткий диск сервера.

Ниже приведен листинг вводной части функции **radiation** с изменениями, которые необходимо произвести, чтобы удовлетворить перечисленным требованиям. Для входной структуры мы используем название `ins`, что является сокращением слов «input structure». В завершение своей миссии расчетная функция должна сформировать текстовую строку `rs`. Ее название происходит от слов «return stream», что буквально означает «возвращаемый поток». Этот поток направляется непосредственно веб браузеру клиента, минуя промежуточную запись на жесткий диск сервера, однако при локальной отладке приложения возвращаемое значение `rs` можно использовать, например, для сохранения на диск компьютера в виде HTML файла.

```
function rs=radiation(ins)
```

```
% Диаграмма направленности излучения частицы.
```

```
% Веб вариант.
```

```
% Считываем исходные данные из структуры ins
```

```
beta    = str2double(ins.beta);
```

```
alpha   = str2double(ins.alpha)*pi/180;
```

```
azimuth = str2double(ins.azimuth);
```

```
elevation = str2double(ins.elevation);
```

```
k       = str2double(ins.k);
```

```
% =====
```

Последующий текст функции **radiation** не нуждается в правке, за тем исключением, что необходимо запомнить дескриптор рисунка для дальнейшей записи рисунка в графический файл. При этом полезно сделать рисунок «невидимым», чтобы сэкономить ресурсы сервера на открытие окна с рисунком, которое пользователь Интернета все равно не мог бы увидеть.

```
% Открываем невидимое графическое окно.
```

```
f = figure('visible','off');
```

```
hold on;
```

В самом конце текста функции нужно добавить еще несколько строк, чтобы сформировать возвращаемую текстовую переменную `rs`, а также записать сгенерированное изображение на жесткий диск компьютера. Однако прежде полезно выполнить еще несколько подготовительных мероприятий по обработке входных параметров.

Рекомендуемая практика программирования для Интернета предполагает, что введенные данные должны пройти хотя бы минимальную проверку перед их использованием. В разделе 4 мы кратко обсудим способы проверки входных данных на стороне клиента. Однако такая проверка не всегда возможна. Злонамеренный пользователь имеет возможность обойти проверку на стороне клиента, введя входные параметры непосредственно в адресной строке веб браузера. Поэтому еще одну проверку необходимо делать на стороне сервера, в тексте программы MATLAB. Как минимум, необходимо убедиться, что все обязательные параметры заданы, а числовые значения лежат в допустимых пределах. Ниже приведен усовершенст-

вованный вариант вводной части функции `radiation` с проверкой входных данных.

```
function rs=radiation(ins)
```

```
% Диаграмма направленности излучения частицы.
```

```
% Веб вариант №2.
```

```
% Проверяем исходные данные из структуры ins.
```

```
es.count = 0;
```

```
if( isfield(in, 'alpha') )
```

```
alpha = str2double(in.alpha);
```

```
if( ~isnumeric(alpha) | alpha < 0 | alpha > 180)
```

```
    es.count = es.count+1;
```

```
    es.description{es.count} = ['Значение alpha='...
```

```
    in.alpha ' вне допустимого интервала.'];
```

```
else
```

```
    alpha    = alpha*pi/180;
```

```
end;
```

```
else
```

```
    es.count = es.count+1;
```

```
    es.description{es.count} = ['Параметр alpha не '... 'задан.'];
```

```
end;
```

Для краткости мы ограничились проверкой одного параметра, но аналогичным или иным способом следует проверить значения всех входных параметров. Информация о выявленных ошибках здесь сохраняется в структуре `es`. При обнаружении ошибки увеличивается счетчик ошибок `es.count`, а пояснительный текст записывается в элемент массива `es.description`. Если после проверки всех параметров значение `es.count` будет больше нуля, выполнение задачи следует прекратить, сформировав из структуры `es` возвращаемую текстовую переменную `rs`. Эту задачу решает функция **getError**, используемая всеми учебными задачами, которые размещены на сайте [MATLAB.tutornet.ru](http://MATLAB.tutornet.ru):

```
function rs = getError( es )
```

```
TXT = '';
```

```
for loop=1:es.count
```

```
    TXT = [TXT '<LI>' es.description{loop} '</LI>'];
```

```
end
```

```
rs = [sprintf('Content-type: text/html\n\n')...
```

```
    '<html><HEAD>
```

```
    '<META HTTP-EQUIV="Content-Type" '...
```

```
    'CONTENT="text/html; '...
```

```
    'charset=windows-1251"></HEAD><body>'...
```

```
    '<p>Ошибки MATLAB:</p>'...
```

```
    '<OL>' TXT '</OL>' ...
```

```
    '</body></html>'];
```

```
% end of getErrorMessage
```

Здесь следует обратить внимание, что возвращаемая строка `rs` должна начинаться с заголовка `'Content-type: text/html'`, который служит указа-

нием на тип возвращаемого потока, а именно на то, что это текстовый поток.

Аналогичным образом текстовый поток с результатами расчета формируется функцией **getHtml**. Она используется в том случае, когда проверка входных параметров не выявила ошибок. Выбор функции, возвращающей текстовый поток **rs**, производится в соответствии со значением счетчика **es.count**. Таким образом, в заключительную часть расчетной функции **radiation** в первом приближении достаточно добавить всего несколько строк:

```
% =====
if( es.count )
    rs = getError( es );
else
    rs = getHtml( ins, gs );
end;
% end of radiation
```

Как видно из приведенного текста, помимо входной структуры, функции **getHtml** нужно передать еще один параметр **gs**. Это еще одна структура. Она содержит информацию о графическом изображении, сформированном расчетной функцией. Для формирования структуры **gs** мы используем общую для всех приложений функцию **getGS**, которая описана в следующем разделе.

Передача структур в аргументах функций чрезвычайно продуктивна, так как позволяет легко решить проблемы, возникающие при обновлении версий функций, как вызываемых, так и вызывающих. При передаче структур между функциями обычно удается обеспечить обратную совместимость версий и параллельную работу как старых, так и обновленных приложений, что чрезвычайно важно для веб приложений, поскольку они должны непрерывно находиться в работоспособном состоянии. При использовании общих для сайта библиотечных функций типа **getError**, **getHtml**, **getGS** или **saveBitmap** (см. ниже) разработчик конкретной задачи освобождается от забот по адаптации своего приложения для размещения на сайте. При обновлении общих функций мы стремимся обеспечить обратную совместимость версий без изменения способа вызова таких функций, неизбежного, например, при изменении числа аргументов. Это позволяет вводить в действие новые версии без единовременного обновления всех приложений. При этом расширение возможностей общих функций обеспечивается путем добавления новых полей к существующим структурам и инкапсуляции выбора сценария, соответствующего конкретной версии, в библиотечную функцию.

### 3. Подготовка графического вывода

Для ускорения отклика веб сервера на запрос веб клиента на сайте MATLAB.tutor.net.ru используется кэширование сгенерированных рисунков на жесткий диск компьютера, на котором установлен веб сервер. После проверки входных параметров производится вызов функции **getGS**. Она определяет, имеется ли в КЭШе (т. е. на диске компьютера) файл с графическим изображением, соответствующим набору входным параметров. Функция **getGS** возвращает структуру **gs**:

```
gs = getGS( ins );
```

Возвращаемая структура содержит имя графического файла с расширением (**gs.filename**) и без расширения (**gs.name**), его тип (**gs.type**), признак наличия файла в КЭШе (**gs.isexist**) и еще ряд полей. Имя файла (**gs.filename**) вычисляется на основе входной структуры путем конкатенации имен полей и их значений, после чего базисная часть имени (т. е. имя файла без расширения) шифруется по алгоритму MD5:

```
% Вычисляем имя файла из входной структуры.
```

```
names = fieldnames(ins);
```

```
for k=1:length(names)
```

```
    field = names{k,1}{:};
```

```
    value = getfield(in, field);
```

```
    switch field
```

```
        % Игнорируем параметры, не влияющие на рисунок.
```

```
        case {'mlmfile', 'mldir', 'mlid'}
```

```
        case {'format'}
```

```
        case {'userAgent', 'host', 'hash', 'search'}
```

```
        case {'availHeight', 'availWidth', 'bufferDepth'}
```

```
        case {'colorDepth', 'height', 'width'}
```

```
        case {'cpuClass', 'platform', 'mediabox'}
```

```
        otherwise
```

```
            % Добавляем параметр и значение к имени.
```

```
            name = [name '(' field value ')'];
```

```
    end % switch
```

```
end % for
```

```
% Шифруем имя файла.
```

```
name=md5(name);
```

Шифрование гарантирует, что имя файла не превышает заданное ограничение по длине имени (64 байта) при том, что два идентичных набора значений параметров всегда отображаются на одну и ту же зашифрованную строку, используемую в качестве имени файла (**gs.name**). Вычислив имя файла, функция **getGS** проверяет его наличие на жестком диске компьютера. Если файл найден, полю **isexist** возвращаемой структуры присваивается значение 1. Анализируя значение **gs.isexist**, расчетная функция должна либо произвести новый расчет, либо использовать имеющийся графический файл **gs.filename**. В нашем примере с построением диаграммы направленности функцией **radiation** достаточно поместить часть тек-

ста, выделенного строками, состоящими из знаков равенства, внутрь условия if:

```
gs = getGS( ins );
if( gs.isexist )
% =====
...
% =====
end; % end of if( gs.isexist )
```

При вычислении имени графического файла функция **getGS** помимо обязательных параметров задачи, проверяет также наличие во входной структуре параметров `ins.format` и `ins.quality`, которые определяют выбор графического формата и качества растрового изображения. В настоящее время параметр `ins.format` может иметь значения 'gif' (или 1), 'mpeg' (или 'mpg', или 2), 'jpeg' (или 'jpg', или 3) и 'png' (или 4). Параметр `ins.quality` может иметь значения 1, 2 или 3. Меньшее значение `ins.quality` соответствует меньшему размеру графического файла, большее значение `ins.quality` соответствует улучшению качества изображения. Если параметр качества `ins.quality` не задан, выбирается оптимальное соотношение качества и размера, т. е. значение 2. Если не задан формат изображения, т. е. поле `format` отсутствует во входной структуре `ins`, предполагается, что следует выбрать формат jpeg. Значения формата и качества, используемые по умолчанию, можно также указать соответственно во втором и в третьем аргументах функции **getGS**. Например, чтобы в приложениях, которые генерируют видеоклипы, по умолчанию использовался формат mpeg с максимальным сжатием файла, следует выбрать следующий способ вызова функции **getGS**:

```
gs = getGS( ins, 'mpeg', 1 );
```

Вне зависимости от наличия параметров `format` и `quality` во входной структуре `ins`, возвращаемая структура `gs` всегда содержит эти поля, причем `gs.format` всегда имеет числовое значение, даже если входной параметр `ins.format` или значение по умолчанию заданы в текстовом виде. Это упрощает последующее использование структуры `gs`.

При типовой внутренней структуре каталога приложения, описанной в статье [2], разработчик задачи может не заботиться о размещении вновь сгенерированного изображения на диске компьютера, так как эту операцию также производит общая функция **saveBitmap** (в случае статического изображения) или **saveMovie** (в случае видеоклипа). Поскольку в примере с диаграммой направленности генерируется статическое изображение, мы ограничимся примером с использованием функции **saveBitmap**. Ее следует вызвать непосредственно перед генерацией возвращаемого текстового потока `rs`. Так что в окончательном виде заключительная часть расчетной функции должна выглядеть следующим образом:

```
% =====
saveBitmap( f, gs );
```



```
end; % end of if( gs.isexist )
```

```
if( es.count )
    rs = getError( es );
else
    rs = getHtml( ins, gs );
end;
% end of radiation
```

Напомним, что *f* — это дескриптор графического окна, созданного командой **figure** (см. раздел 2).

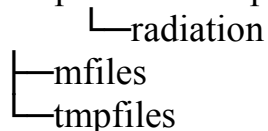
Функция **saveBitmap** сохраняет изображение, сформированное расчетной функцией в окне *f*, в формате *gs.format* с качеством, определяемым параметром *gs.quality* в файле с расширением *gs.type*. Приведем часть текста функции **saveBitmap**, отвечающую за сохранение изображения в формате jpeg:

```
print(f, '-djpeg', '-r0', gs.name);
if gs.quality == 1
    tmp = imread(gs.name, 'jpg');
    imwrite(tmp, gs.filename, 'jpg', 'Quality', 25);
elseif gs.quality == 3
    tmp = imread(gs.name, 'jpg');
    imwrite(tmp, gs.filename, 'jpg', 'Quality', 100);
end;
```

Для понимания приведенного фрагмента полезно иметь в виду, что полное имя файла *gs.filename* функцией **getGS** определено как [*gs.name* '.' *gs.type*].

Перед сохранением изображения в файл функция **saveBitmap**, принудительно завершает все операции с графическими объектами, используя встроенную функцию MATLAB **drawnow**.

При выборе каталога, куда функция **saveBitmap** сохраняет файл с графическим изображением, предполагается, что каждое приложение размещено в отдельном каталоге, внутри которого имеется дочерний каталог *mfiles* с текстами функций MATLAB и дочерний каталог *tmpfiles*, предназначенный для хранения временных файлов, в том числе графических изображений, сгенерированных приложением. Иными словами, каталог рассматриваемого приложения имеет следующую структуру:



Как показано в [2], такая структура каталогов облегчает администрирование веб сервера и позволяет в конечном итоге автоматизировать генерацию веб интерфейса. Рабочим каталогом для приложения MATLAB является каталог *mfiles*. Поскольку полученное изображение необходимо сохранить в «сестринский» каталог *tmpfiles*, функция **saveBitmap** в про-

стейшем случае перед записью графического файла производит переход в каталог хранения временных файлов:

```
cd('../tmpfiles');
```

Чтобы избежать переполнения каталога хранения временных файлов, эта же функция производит очистку каталога, удаляя из него все файлы, срок хранения которых превысил определенный предел. Для очистки используется функция **wscleanup**, входящая в комплект MWS. В настоящее время мы удаляем файлы, хранящиеся более 36 часов:

```
wscleanup('*.*', 36);
```

Такой выбор времени хранения был выбран из соображений удобства при использовании учебных задач в лекционных демонстрациях, поскольку лектор подбирает параметры демонстраций обычно в день перед лекцией.

#### 4. Генерация текстового потока

В завершение своей миссии расчетная функция (в рассматриваемом примере это функция **radiation**) должна сформировать текстовую строку rs. Как показано в [2], в простейшем случае можно ограничиться минимумом, который должен включать HTML тег <img> с атрибутом src, указывающим на виртуальный адрес сгенерированного графического файла на веб сайте:

```
rs = [sprintf('Content-type: text/html\n\n') ...
```

```
'<html><body>' ...
```

```
'</body></html>']
```

Однако на сайте MATLAB.tutornet.ru применяется более продвинутая технология генерации возвращаемого текста. Она инкапсулирована в библиотечную функцию **getHtml**, которая уже упоминалась выше. В общем случае функция **getHtml** способна принимать три аргумента, но только первые два из них обязательны. Третий аргумент может использоваться при отладке задачи. Если указаны все три аргумента, то результат вызова функции

```
rs = getHtml( ins, gs, outfile );
```

помимо передачи в указанную переменную rs дополнительно записывается в файл outfile на жесткий диск компьютера.

Соответствие физического адреса графического файла в файловой системе компьютера его же виртуальному адресу на веб сайте определяется настройками веб сервера. Эта тема детально рассмотрена в статье [2]. Здесь же поясним, что, получив текст, веб браузер клиента обнаружит в нем тег <img> и отправит на веб сервер еще один запрос, затребовав рисунок. Веб сервер в ответ отправит рисунок с жесткого диска компьютера.

На одной веб странице можно разместить несколько рисунков, сгенерированных расчетной программой, но такая возможность не реализуется

ся функцией **getHtml**, поскольку в настоящее время все задачи, размещенные на сайте MATLAB.tutornet.ru, производят построение не более чем одного статического рисунка или видеоклипа. Функция **getHtml** выбирает наиболее подходящий способ внедрения полученного графического или видеоизображения в возвращаемый текстовый поток сообразно типу изображения (gs.format) и типу веб браузера клиента.

Тип веб браузера в настоящее время автоматически определяется HTML формой и передается через скрытый параметр interface. Если клиент использует Internet Explorer версии 5.5 или выше, этому параметру присваивается значение ie. В этом случае возвращаемый текст HTML формируется с учетом расширенных возможностей браузера Internet Explorer. Во всех остальных случаях, в том числе при отсутствии параметра interface в запросе, формируется максимально упрощенный текст HTML, который предположительно должен без ошибок интерпретироваться веб браузерами устаревших типов. MATLAB Web Server не способен самостоятельно определить тип веб браузера клиента (в отличие от большинства других серверных продуктов), что, вероятно, следует признать определенным недостатком MWS. Поскольку в настоящее время Internet Explorer занимает доминирующее положение на рынке веб браузеров, мы не предпринимали попыток гарантировать определение типа веб браузера при любом способе вызова приложения через веб браузер, а не только через HTML форму (хотя это гипотетически возможно). Вместо этого мы максимально расширили функциональные возможности возвращаемого текста HTML с помощью клиентских скриптов JavaScript. Эти скрипты определяют тип веб браузера и, если позволяют возможности браузера, производят дополнительные настройки возвращаемого изображения. Например, при использовании Internet Explorer'a по окончании загрузки изображения производится его автоматическое масштабирование под размер окна браузера, а ход вычислений и загрузки отображается на экране клиента.

Помимо тега `<img>` с атрибутом src, для отображения сгенерированного изображения используется также тег `<img>` с атрибутом dynsrc и тег `<embed>`. Выбор тега и атрибутов основывается на анализе параметров gs.format и ins.interface. Например, `<img>` с атрибутом dynsrc выбирается функцией **getHtml** для анимированного изображения в формате gif (gs.format==1) при условии, что ins.interface=='ie'. В остальных случаях для видеоизображения выбирается тег `<embed>`, а для статического рисунка (gs.format==3 | gs.format==4) — тег `<img>` с атрибутом src.

## **5. Выбор веб интерфейса**

Существует четыре основных варианта встраивания приложения MATLAB в HTML страницы.

Во-первых, приложение можно вызвать из адресной строки веб-браузера. Например, рассмотренное выше приложение реально размещено на сайте MATLAB.tutornet.ru, так что диаграмму направленности, изображенную на Рис. 1, можно получить, набрав в адресной строке следующий запрос:

`http://MATLAB.tutornet.ru/cgi-bin/matweb.exe?mlmfile=radiation&beta=0.7`

`&alpha=30&elevation=15&azimuth=30&k=60`. Если пользователь совершит ошибку в запросе, он получит исчерпывающую информацию, которая позволит легко ее устранить. Пример сообщения об ошибке приведен на Рис.2. Мы используем данный способ вызова для проверки приложений относительно их устойчивости к злонамеренным действиям посетителей сайта.

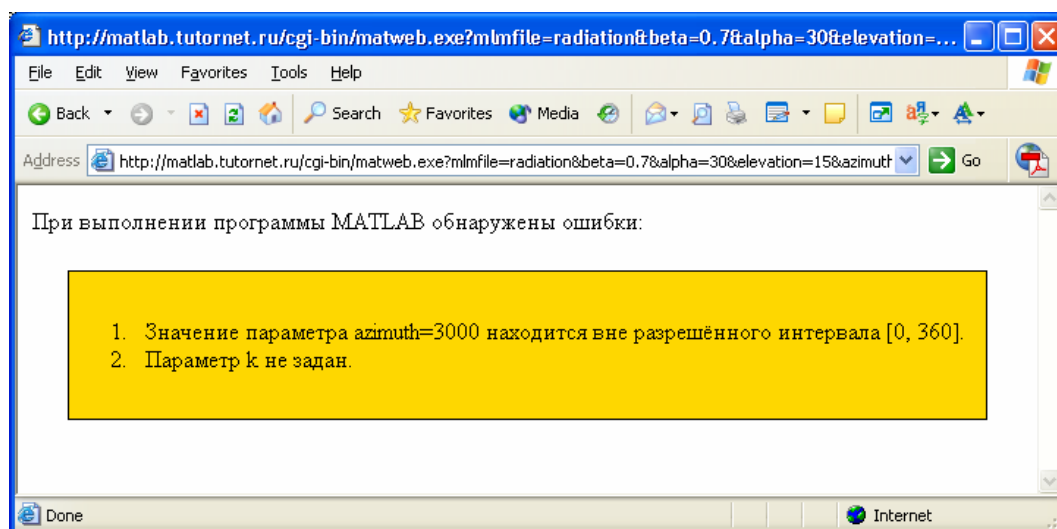


Рис. 2. Сообщение об ошибке во входных параметрах.

Во-вторых, запрос на выполнение приложения можно оформить в виде гиперссылки. Для этого в текст произвольной HTML страницы достаточно вставить тег `<a>`, указав параметры запроса в его атрибуте `href`. Например:

```
<a href="http://MATLAB.tutornet.ru/cgi-bin/matweb.exe?mlmfile=radiation&beta=0.7&alpha=30&elevation=15&azimuth=30&k=60">Излучение релятивистской частицы</a>
```

При клике указателем «мышки» по гиперссылке, которая в окне браузера будет в данном случае отображена в виде текста «Излучение релятивистской частицы», произойдет переход на страницу с результатом работы приложения. Добавив атрибут `target="_blank"`, можно указать, чтобы результат расчета всегда выводился в новое окно, что особенно полезно для сравнения нескольких расчетов.

В-третьих, результат работы приложения можно встроить в любую HTML страницу с помощью тега `<iframe src="...">`. При этом родительская страница может находиться на другом сайте, а его посетители могут

даже не догадываться, что «картинка позаимствована». Необходимы специальные меры защиты от подобного заимствования, если оно нежелательно.

Наконец, параметры задачи можно задавать, формируя их с помощью HTML формы. Только этот, четвертый, способ позволяет обеспечить истинную интерактивность взаимодействия пользователя с расчетной программой. HTML форма формирует запрос примерно в том виде, как он был выше продемонстрирован в примере с адресной строкой, однако обычно параметры запроса передаются так, что пользователь не видит их в адресной строке.

Изложенные выше принципы адаптации приложения MATLAB к размещению на веб сайте позволяют одинаково успешно использовать это приложение с любым веб интерфейсом.

## **6. Продвинутый интерфейс с использованием HTML форм**

Начальные сведения о разработке HTML формы можно почерпнуть из любого учебника по языку разметки гипертекстов HTML. Документация по MWS таких сведений почти не содержит. На сайте MATLAB.tutor.net.ru помимо своей основной функции отправки данных на сервер, HTML форма делает много других полезных вещей.

Во-первых, она осуществляет предварительную проверку введенных данных перед их отправкой на сервер (проверку на стороне клиента). Мы используем как пассивную проверку на стороне клиента (с помощью запрета ввода очень длинных текстовых строк в полях ввода), так и активную проверку (с помощью скриптов JavaScript).

Во-вторых, HTML форма накапливает информацию о скорости соединения между веб клиентом и веб сервером. На основе этой информации производится выбор параметра качества изображения (quality). Пользователю также предоставлена возможность ручного выбора этого параметра.

В-третьих, HTML форма собирает информацию о свойствах веб браузера клиента, размерах экрана и его разрешающей способности, а также о глубине цвета. В настоящее время эта информация реально не используется, однако в будущем она может быть использована при выборе параметров изображения, генерируемого задачей.

В четвертых, HTML форма реализует определенный сценарий смены изображения в информационном кадре задачи. На Рис. 3-5 он находится слева от панели ввода параметров.

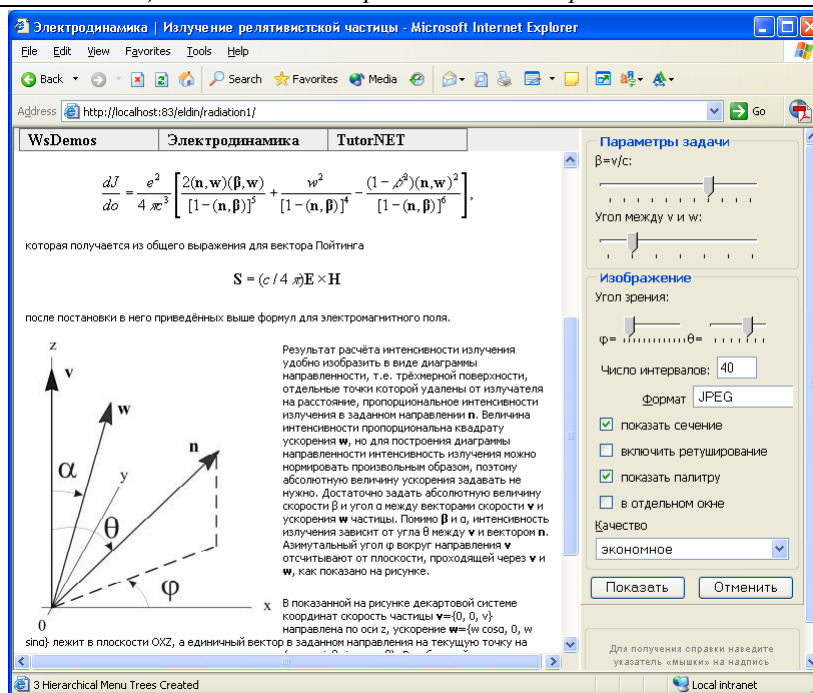


Рис. 3. Окно задачи перед началом расчета.

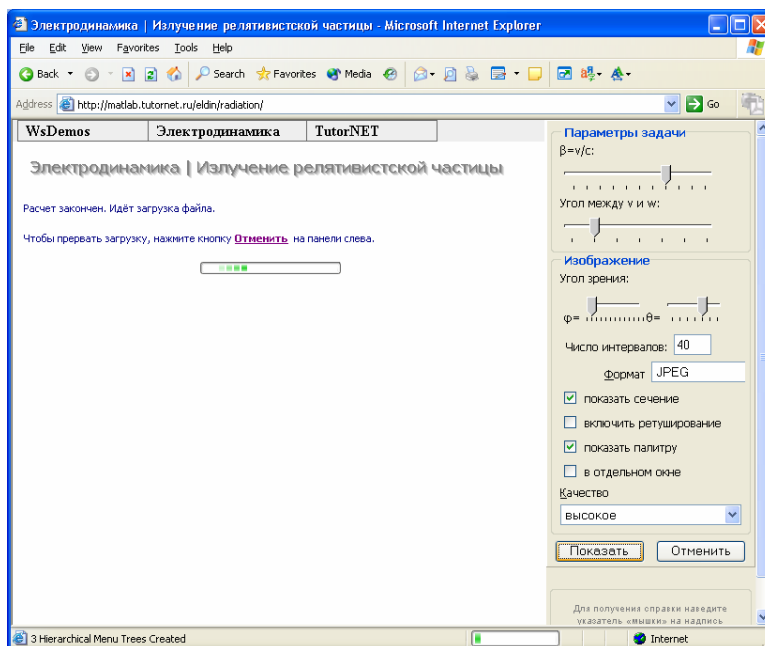


Рис. 4. Окно задачи при загрузке результата расчета.

Первоначально в информационный кадр загружаются сведения о задаче и расчетной модели (Рис. 3). После нажатия пользователем кнопки «Показать» в панели ввода параметров форма вводит небольшую задержку (0,5-1 сек.) перед отправкой данных на сервер, достаточную чтобы загрузить в информационное окно сообщение о предполагаемом времени расчета (не показано). После завершения расчета в информационном окне в варианте для Internet Explorer появляется сообщение о начале загрузки изо-

бражения (Рис. 4). После завершения загрузки данное сообщение заменяется сгенерированным изображением, которое предварительно масштабируется под размер информационного кадра.

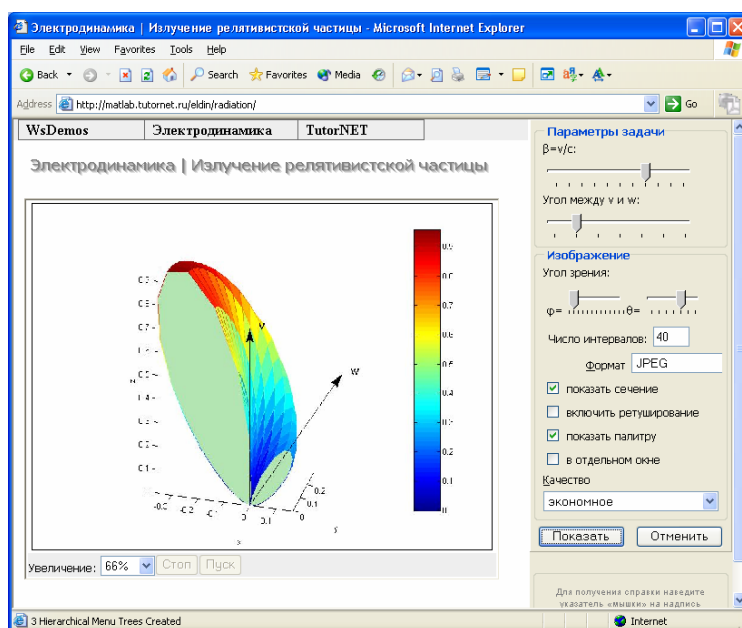


Рис. 5. Окно задачи с результатом расчета.

Все используемые приложениями HTML файлы и скрипты JavaScript автоматически создаются для каждой задачи разработанным нами генератором веб интерфейса. Заметим, что этот же генератор формирует блок проверки входной структуры в m-файлах приложений MATLAB, поэтому реально разработчик задачи может всего лишь обозначить место в тексте m-файла, где такая проверка должна быть выполнена.

Исходная информация для генератора интерфейса извлекается из описания задачи на расширяемом языке разметки XML. Использование единого источника информации с описанием задачи обеспечивает согласованность проверки входных параметров на стороне клиента и на стороне сервера. Использование генератора веб интерфейса освобождает разработчика задачи от рутинной работы по разработке HTML файлов, собственно составляющих веб интерфейс приложения.

В настоящее время каждое приложение использует как минимум 6 HTML файлов:

1. default.htm — главный файл веб интерфейса, определяющий расположение, структуру и размеры панели ввода параметров и информационного окна;
2. info.htm — файл описания задачи, содержащий ее краткое описание и инструкции для пользователя по выбору значений входных параметров;
3. panel.htm — файл панели ввода параметров для веб браузеров с ограниченными возможностями;

Все HTML файлы создаются генератором веб интерфейса из XML описателя задачи путем последовательного наложения на него XSL стилей. Генератор веб интерфейса написан на языке JavaScript и работает под управлением Windows Script Host, являющегося штатным компонентом операционной системы Windows. На Рис. 6 представлен вид окна справки генератора веб интерфейса.

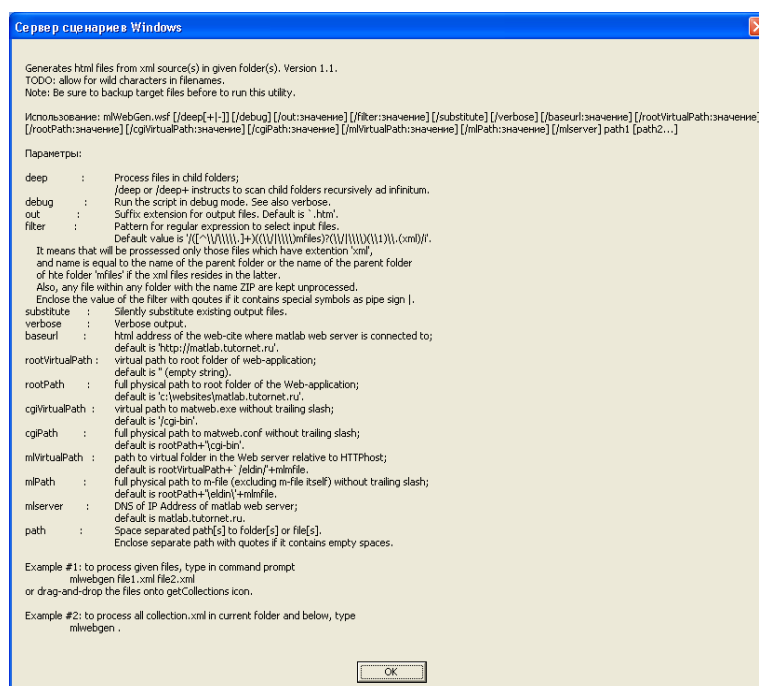


Рис. 6. Генератор веб интерфейса.

Генератор может работать как в оконном, так и в консольном режиме. После создания HTML файлов он производит проверку конфигурации настройки веб сервера и сервера MATLAB. В частности он устанавливает запрет на доступ анонимного пользователя Интернета к исходному тексту приложений MATLAB, размещенных непосредственно на веб сервере.

Практическое применение генератора веб интерфейса чрезвычайно просто. Для обновления интерфейса всех задач, размещенных на сайте, достаточно в проводнике Windows Explorer захватить указателем «мышки» каталог со всеми приложениями (или группы каталогов с выбранными приложениями) и «опустить» его на иконку генератора. Можно также использовать вызов генератора через bat-файл, если требуется изменить параметры генерации, используемые по умолчанию. Это позволяет в течении нескольких секунд вносить отложенные изменения в интерфейс всех задач,



размещенных на сайте. Необходимой предпосылкой успешной работы генератора является использование стандартизированной структуры каталогов приложений и применение описанной в настоящей статье библиотеки функций ввода-вывода в тексте расчетных функций.

## **7. Заключение**

Настоящая работа является логическим продолжением статьи [2], где изложены первоначальные сведения об использовании и настройке MWS. Мы рассчитываем, что описанный выше опыт авторов по разработке учебных задач по курсу классической электродинамики, размещенных на авторском сайте MATLAB.tutornet.ru (<http://MATLAB.tutornet.ru/eldin/>) и на сайте физического факультета НГУ (<http://www.phys.nsu.ru:8000/eldin/>), будет полезен не только преподавателям университетов, которые стремятся расширить свой арсенал средств обучения студентов, но и научным работникам, которые работают в составе коллективов, объединяющих усилия нескольких исследовательских организаций.

Работа выполнена при финансовой поддержке Президента Российской Федерации (грант для поддержки ведущих научных школ № НШ-229.2003.2) и Министерства общего и профессионального образования РФ (программа «Интеграция науки и высшего образования России на 2002-2006 годы», государственный контракт №Ф0051).

## **Литература**

1. MATLAB Web Server — документация к программному обеспечению MATLAB.— MathWorks, Inc., 2002.
2. Котельников И. А., Черкасский В. С. MATLAB Web Server: вычисления в Интернете // Exponenta Pro. Математика в приложениях.— 2004.— №1.
3. Ландау Л.Д., Лифшиц Е.М. Теория поля.— М.: Наука, 1988.
4. Котельников И. А., Матвеев А. Н., Черкасский В. С. Разработка обучающих программ с использованием MATLAB Web Server // Тез. докл. Всероссийской науч. конф. «Проектирование научных и инженерных приложений в среде MATLAB» (28–29 мая 2002). — М.: ИПУ РАН, 2002.— С.207.
5. Котельников И. А., Матвеев А. Н., Черкасский В. С. Разработка и использование моделирующих программ для дистанционного и самостоятельного обучения с использованием MATLAB Web Server. // Тез. докл. международ. науч.-метод. конф. «Новые информационные технологии в университетском образовании» (Новосибирск: СибГУТИ, 23–24 сентября 2003).— Новосибирск: СибГУТИ, 2003.— Т.1.— С.92–94.

УДК 53.072:53:372.8

## **ОПЫТ ИСПОЛЬЗОВАНИЯ СРЕДЫ БЫСТРОЙ РАЗРАБОТКИ ПРИЛОЖЕНИЙ MATLAB В УЧЕБНОМ ПРОЦЕССЕ НА ФИЗИЧЕСКОМ ФАКУЛЬТЕТЕ НГУ**

*Коткин Г. Л., Черкасский В. С.,*

*Новосибирский государственный университет, Новосибирск,*

*e-mail: cherk@phys.nsu.ru*

*Котельников И. А.*

*Институт ядерной физики им. Будкера СО РАН, Новосибирск,*

*e-mail: I.A.Kotelnikov@inp.nsk.su*

Сегодня никого не надо убеждать в важности использования информационных технологий (ИТ) как для целей обучения в конкретной предметной области (лекционные демонстрации, компьютерные и лабораторные практикумы, решение задач с помощью тех или иных компьютерных систем), так и для обучения способам владения самими этими средствами (например, универсальными языками программирования). Опыт такого использования ИТ в образовании на физическом факультете НГУ насчитывает более 25 лет [1]. Зачастую каждая из перечисленных выше задач решается своими средствами с использованием разных языков программирования, различных технологий и даже разных операционных систем. Настоящий доклад посвящен описанию использования среды быстрой разработки приложений MATLAB для решения большей части этих задач.

Первая часть доклада посвящена описанию практикума «Компьютерное моделирование физических процессов с использованием MATLAB», который преподается на физическом факультете НГУ во втором семестре первого курса после ознакомительного курса «Введение в информационные технологии» в первом семестре. Этот практикум является комплексным — наряду с изучением языка программирования MATLAB большое внимание в курсе уделяется собственно задачам моделирования.

Вторая часть доклада посвящена описанию опыта использования среды быстрой разработки приложений (СБРП) MATLAB для создания «закрытых» моделирующих задач, которые используются для изучения конкретных физических явлений (на примере электродинамики, изучаемой студентами в третьем и четвертом семестрах) как в режиме лекционных демонстраций, так и в режиме самостоятельной работы студентов. Причем эта работа может осуществляться как в локальном режиме (с установленной на компьютере СБРП MATLAB), так и через Интернет.

## **Моделирование физических явлений с помощью MATLAB**

Для обучения компьютерному моделированию физических процессов необходимо научить студента

1. формулировать математическую постановку физической задачи и находить методы ее решения;
2. «программировать на компьютере», т. е. использовать среду разработки приложений для ввода и отладки программы и поиска необходимой информации по языку программирования в справочной системе СБРП;
3. представлять получаемые результаты в наглядной форме, которая позволяет лучше понять происходящие процессы.

При этом желательно, чтобы у обучаемого не угас интерес по мере продвижения по всем этим этапам, а это возможно только тогда, когда на каждом шаге возникают относительно интересные задачи, которые удастся решать уже знакомыми средствами.

На этом пути нами было принято решение не разделять все эти три стороны процесса моделирования. При стандартном подходе студент изучает три отдельные дисциплины (уравнения математической физики и численные методы их решения, программирование и собственно физику), причем они редко бывают связаны между собой. Такие же вопросы, как представление результатов расчетов в наглядной форме и их анализ вообще не встречается ни в каком курсе. Существенным отличием ныне существующего практикума «Компьютерное моделирование физических процессов с использованием MATLAB» от его предшественников [1-3] является переход от универсального языка программирования (как Фортран или Паскаль) к СБРП MATLAB фирмы MathWorks, созданной для решения инженерных и научных задач. Выбор системы MATLAB был осуществлён в ходе дискуссий с участием широкого круга преподавателей в 1997-1998 годах. Принятое решение было мотивировано тем, что язык MATLAB является интерпретирующим языком высокого уровня, а СБРП MATLAB имеет богатую библиотеку встроенной графики и легко создаваемый графический интерфейс пользователя. Не последнюю роль в предпочтении системе MATLAB перед конкурирующими системами научных расчетов, таких как MathCAD, Mathematica и Maple, сыграло то обстоятельство, что среди преподавателей уже тогда были ярые приверженцы MATLAB, один из которых (В.Ч.) является автором данной статьи. Наш опыт подтверждает известное правило, что при наличии примерно равных программных продуктов при выборе средства обучения студентов предпочтение следует отдавать тому продукту, который имеет наибольшее число приверженцев среди преподавателей, так как хорошо преподавать конкретный язык программирования могут только его энтузиасты.

В ранних вариантах компьютерного практикума на физическом факультете НГУ при работе на Паскале авторы курса сами разрабатывали

графический интерфейс пользователя, а в СБРП MATLAB разработка такого интерфейса по силам даже новичку. Таким образом, одна из задач данного курса — научить студента пользоваться высокоуровневыми средствами программирования и осваивать предоставленные средства.

### ***Обучение программированию и особенности графического интерфейса***

В настоящее время курс состоит из 8 лекций (16 часов) и 48 часов практических занятий в компьютерном классе. С первого занятия студенты начинают работать с исходными текстами программ, приготовленными заранее для этого курса, что позволяет им сразу познакомиться с языком MATLAB (не в полном объеме, зато сразу же на уровне «разговорного»). Эти программы представляют собой с одной стороны каркас будущей программы студента, а с другой стороны — вполне работоспособные программы, которые содержат как расчетную часть простейшего варианта соответствующей модели (см. далее), так и уже созданный графический вывод. Например, простейший вариант расчета и вывода на экран графика биений при сложении двух колебаний с близкими частотами имеет вид:

```
function r=run_beats()
% Задание параметров задачи
a1=1; a2=1.1;
w1=1.0;w2=1.2;
t0=0;tm=25;
N=500;
T=tm-t0;
dt=T/N;
% Основной блок вычислений
t=t0:dt:tm;
%Вычисление функции «биений»
y=a1*cos(w1*t)+a2*cos(w2*t);
% Отрисовка полученного графика
plot(t,y);
```

Эта функция вполне работоспособна, и на ее примере мы далее продемонстрируем создание графического интерфейса, как это предлагается сделать студенту.

При дальнейшей работе (по мере надобности для решения конкретной задачи) обучаемые знакомятся с такими элементами языка как циклы **While** и **For**, функциями, использованием функций в качестве аргументов других функций, условными операторами **If**, **If Else** и **Case**. Далее студенты осваивают со стандартными функциями графического вывода типа **Plot**, **Axis**, а также с технологией анимационного вывода (путём изменения свойств объекта **Line**). Например, в задаче о биениях, приведенной выше, студенту предлагается модифицировать программу так, чтобы вывод осуществлялся в несколько окон на одном рисунке, причем в каждом

из окон параметр отношения частот был другим. При этом студенты знакомятся с функцией **subplot**, а также с функцией вывода заголовка **title**.

В результате такого обучения, в каждом случае нацеленного на решение практической задачи, студент осваивает основы языка MATLAB (за исключением сложных типов данных и работы с файлами данных), а также приобретает опыт использования пользовательских функций графического интерфейса, которые предоставляет разнообразные услуги по взаимодействию с процессом счета и графическим выводом. Хорошо успевающие студенты, желающие совершенствовать свои знания в программировании, могут модифицировать эти функции для решения своих задач.

### ***Методика решения физических задач***

За время обучения студент обязан решить четыре задачи из обязательного списка (по выбору преподавателя) и одну задачу по собственному выбору либо из предложенного списка (15 задач), либо свою собственную задачу по согласованию с преподавателем. В список обязательных задач входят:

1. ознакомительные задачи «Биения» и «Фигуры Лиссажу»,
2. одномерные колебания «Маятник»,
3. движение в центральном поле «Планеты»,
4. случайные блуждания «Диффузия»,
5. броуновское движение,
6. кинетика газа дисков,
7. ослабление пучка частиц при прохождении через вещество.

Список задач для самостоятельного решения значительно шире и постоянно пополняется преподавателями с учётом их собственных интересов и направления научных исследований. Все задачи из обязательного списка включены в учебное пособие [4], в котором приводится физическая постановка задачи, основные методы её решения, особенности реализации используемых алгоритмов, а также текст каркасной программы. Текст каркасной программы (подобно приведенной выше программе «Биения») доступен студенту в электронном виде, что позволяет ему сразу запустить задачу на счёт и получить результаты в самой простой реализации модели. Руководствуясь списком заданий и вопросов, приведённых в пособии, студент далее модернизирует свою программу, чтобы ответить на поставленные вопросы. При этом происходит одновременное углубление понимания физической природы моделируемого явления и освоение методов численной реализации модели и представления результатов. Опыт работы со студентами показывает, что результаты своих численных экспериментов они запоминают навсегда, что помогает в их изучении физики, которая, казалось бы, не является основным предметом изучения в компьютерном практикуме.

### Графический интерфейс в задаче «Биения»

Для проведения серийных расчётов при решении как учебных, так и научных задач удобно вводить исходные данные без перезапуска программы, не меняя текст самой программы. Для реализации такого способа взаимодействия с программой мы разработали интерактивный графический интерфейс. Он позволяет видеть на экране одновременно окно вывода графической информации, «кнопки управления» и окна редактирования данных (рис. 1).

Создание такого интерфейса включает две самостоятельные стадии: разработку внешнего вида интерфейса (сколько и каких окон для графиков, какие кнопки и редактируемые поля и т. д. будут на экране) и связывание этого интерфейса с вычислительной программой.

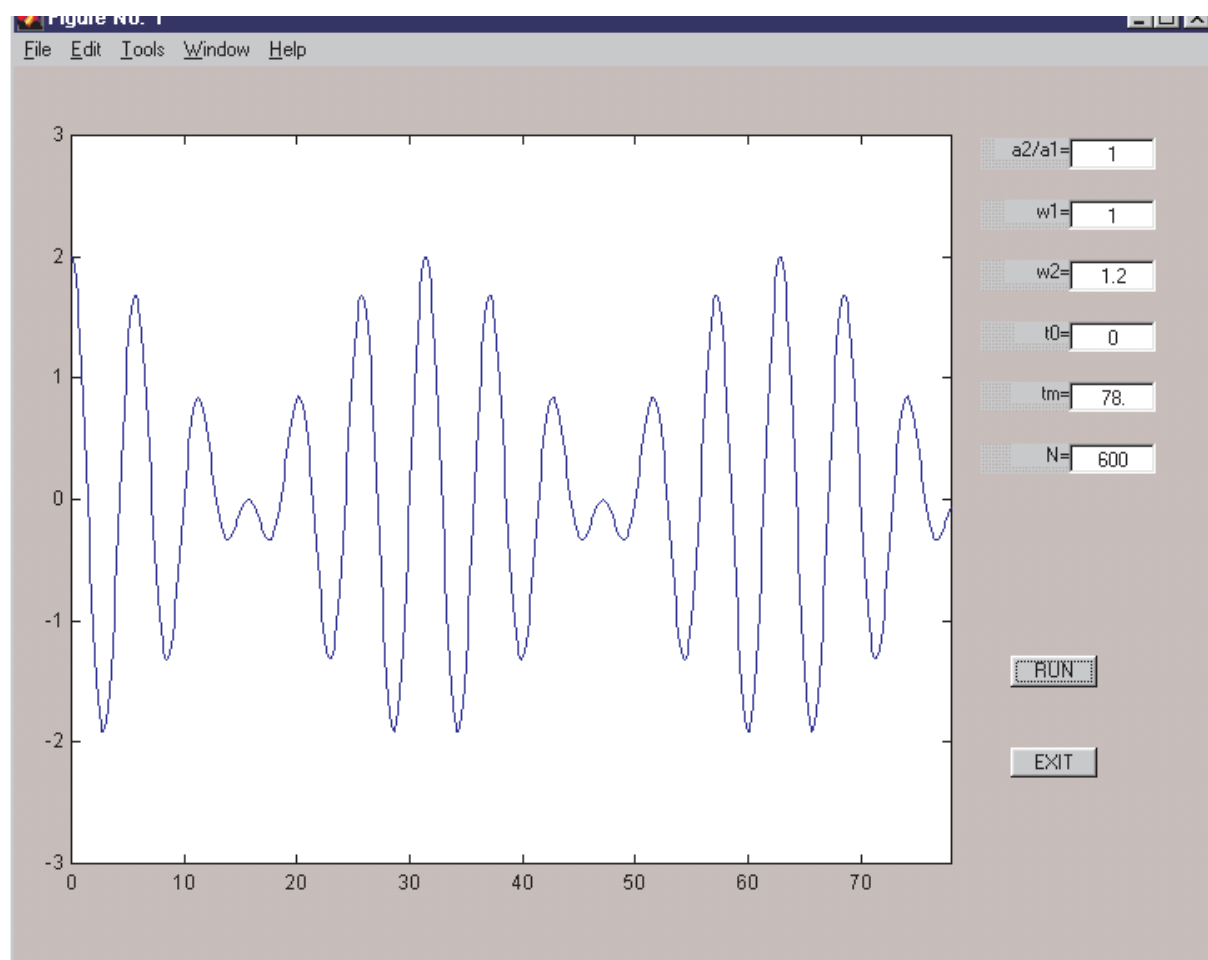


Рис.1. Графический интерфейс для задачи «Биения»/

Созданная с помощью стандартного визуального редактора GUIDE (MATLAB, версия 5.3) функция **Interface\_Window**(APAS, NAMEPAS, RUN\_NAME) была потом доработана «вручную» таким образом, что число полей ввода данных автоматически изменяется в зависимости от длины

массива редактируемых переменных APAS. Вызов этой функции осуществляется из главной программы (или функции без параметров) по следующей схеме

```
clear; % Очистка предыдущего состояния
% Задание начальных значений параметров
a2=1;
w1=1;
w2=1.2;
t0=0;
N=600;
tm=25;
%Создание «транспортной» переменной
APAS(1)=a2;
APAS(2)=w1;
APAS(3)=w2;
APAS(4)=t0;
APAS(5)=tm;
APAS(6)=N;
% Создание имен переменных для вывода
NAMEPAS(1,:)= ' a2/a1';
NAMEPAS(2,:)= ' w1';
NAMEPAS(3,:)= ' w2';
NAMEPAS(4,:)= ' t0';
NAMEPAS(5,:)= ' tm';
NAMEPAS(6,:)= ' N';
```

```
%Вызов функции графического интерфейса
Interface_Window(APAS,NAMEPAS,Run_beats);
```

На втором этапе разработки проекта с графическим интерфейсом студент переделывает расчётную часть программы, заменяя ту её часть, где производится ввод исходных данных. Приведём вводный фрагмент функции Run\_beats с произведёнными изменениями.

```
function r=Run_beats()
% Определение дескриптора текущего рисунка
h0=gcf;
% Извлечение структуры info из свойства Userdata
info=get(h0,'Userdata');
a2=info.APAS(1);
w1=info.APAS(2);
w2=info.APAS(3);
t0=info.APAS(4);
tm=info.APAS(5);
N=info.APAS(6);
```

Функция **Run\_beats** вызывается при нажатии клавиши **RUN**. Она считывает входные параметры из структуры info и реализует основной вычислительный алгоритм. Поскольку эта структура сама является полем структуры, хранящейся в свойстве Userdata текущего рисунка, к ним имеет

также доступ специально написанная функция редактирования **Edit\_beats**, что и делает эту систему интерактивной.

После выполнения вычислительной части функции производится вызов функции рисования **plot**. После вывода графиков программа переходит в режим ожидания действий пользователя, что позволяет ему отредактировать значения входных параметров и повторить расчет, вновь нажав кнопку RUN. Выход осуществляется по нажатию клавиши EXIT, что реализуется функцией **exit\_beats**, которая с помощью встроенной функции **delete(gcf)** удаляет текущий рисунок. Таким образом, используя подготовленные функции студент, а также преподаватель непрограммист могут легко переделать свою вычислительную программу в программу с удобным интерактивным интерфейсом. Описанные функции были разработаны для MATLAB 5.3, и, хотя в MATLAB 6.X технология разработки подобных приложений изменилась, все разработанные функции работают и в новой версии.

### ***Использование MATLAB в курсе электродинамики***

Учебные программы, моделирующие физические явления, которые позволяют представить результаты расчетов в виде графических и анимационных изображений, являются действенным инструментом обучения в арсенале преподавателей естественнонаучных дисциплин. Не подменяя живое общение преподавателя и студента, они дополняют традиционное изложение предмета в виде набора простых моделей, которые можно объяснить «на пальцах», средствами визуализации теоретических построений. Особую ценность представляют программы двойного назначения, которые можно использовать как для обучения студентов на практических занятиях и лекционных демонстраций, так и для самостоятельных занятий обучаемых через Интернет при предварительном ознакомлении с материалом или для закрепления полученных знаний.

### ***Выбор языка программирования***

Использование СБРП MATLAB позволяет быстро разрабатывать и отлаживать локальные приложения с развитым графическим интерфейсом (см. выше). Входящий в состав системы MATLAB пакет MATLAB WEB Server позволяет преобразовывать разработанные учебные задачи в веб приложения, пригодные для размещения в Интернете [5].

При разработке учебных и (или) демонстрационных программ первый вопрос, который приходится решать, — должна ли быть разрабатываемая программа открытой или закрытой. Закрытой программой мы называем такую программу, при использовании которой возможно задание различных параметров, определяющих моделируемый процесс, а сам алго-



ритм производимых вычислений и программа являются неизменными и, как правило, недоступными. Открытой программой мы называем такую программу, которую может видоизменять пользователь для получения тех или иных ответов. Если целью обучения является обучение моделированию (см. выше), то предпочтительной является открытая программа, текст которой доступен пользователю, и он может сам видоизменять ее, решая новые задачи. Если же говорить об обучении предметной области (физике, математике, химии), то предпочтительным является тщательно разработанная закрытая программа с удобным графическим интерфейсом, использование которой не требует знания языка программирования. Достаточно научиться пользоваться такой программой. Такая же программа является предпочтительной при проведении лекционных демонстраций.

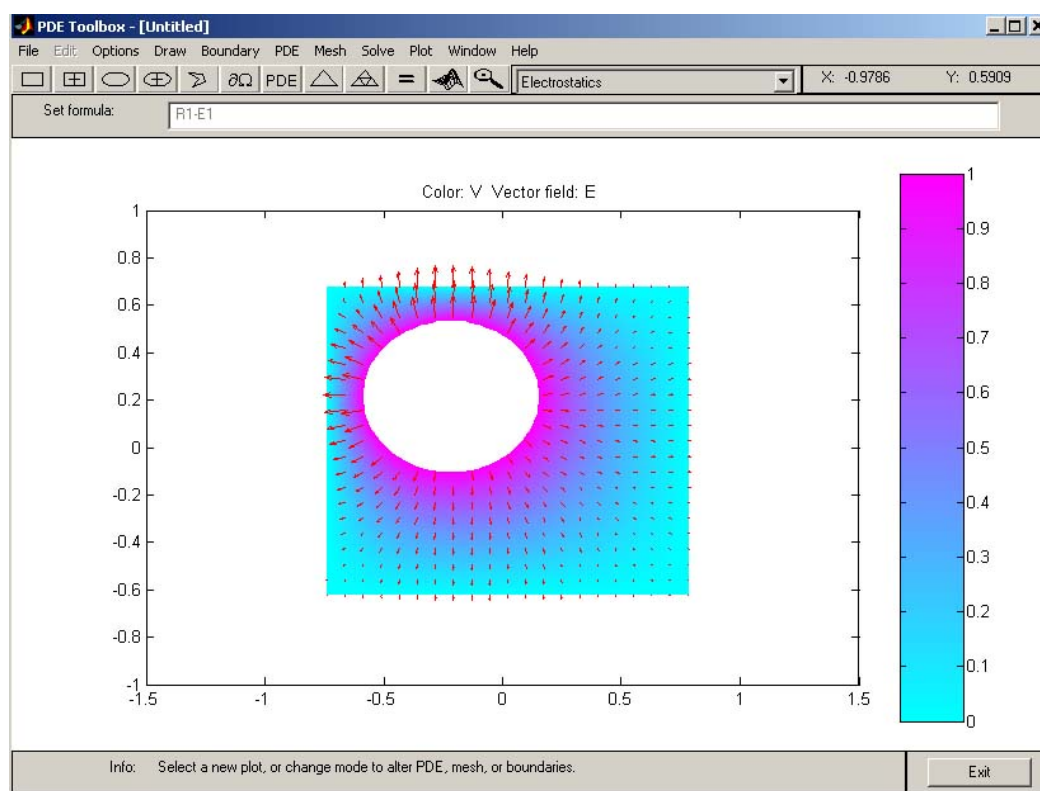


Рис. 2. Стандартный интерфейс пакета PDE для решения задачи электростатики.

Поскольку целью нашей работы было создание моделирующих (обучающих) программ и/или демонстрационных программ по физике (в данном конкретном случае — по электродинамике), мы выбрали вариант закрытой программы, текст которой частично или полностью недоступен пользователю. Один из вариантов разработки такой программы предлагает фирма Mathworks — это пакет PDE (Partial Differential Equations — уравнения в частных производных), который совместно с задачей **pdetool** (см. рис. 2) предоставляет удобные средства решения двумерных задач Лапласа и Пуассона. Фактически, демонстрационная задача **pdetool**, поставляемая

фирмой MathWorks, является образцом для разработки подобных задач. Система имеет удобный графический ввод области решения, интуитивно понятный интерфейс для ввода граничных условий и достаточно развитые средства вывода результатов расчета. Поскольку имеется отдельный пункт меню, позволяющий решать задачи электро- и магнитостатики, то при небольших доделках данный пакет используется для проведения практических занятий по электродинамике в компьютерном классе. В частности, при решении многих задач электростатики, сводящихся к решению уравнения Лапласа и Пуассона, по результатам расчета потенциала необходимо помимо построения картины поля и/или потенциала находить распределение зарядов и токов на граничных металлических поверхностях или на границах раздела сред. Предложенный пакет не содержал таких средств, и пришлось его дорабатывать. При этом, конечно, не хотелось просто скопировать полученное с помощью PDE распределение поля и сетки в заданной области в другую программу на MATLAB'е и там вычислять требуемые величины. Сама же функция вычисления нормальной производной на границы (а именно через эту величину выражается заряд и ток на границе) отсутствует как в стандартных функциях MATLAB, так и в пакете PDE. Мы разработали процедуру вычисления емкости и заряда на поверхности и добавили в меню Solve задачи **pdetool** новый пункт Capacity (Емкость). Оригинальный алгоритм такого расчета нам подсказал (после обращения через форум сайта Matlab.ru) ведущий одного из разделов Шмелев В. Е.

Большинство представляющих интерес демонстраций и учебных задач представляют собой решение динамических задач, поэтому для их представления как локально, так и в Интернете представляется важным вывод анимированных изображений. И хотя в системе MATLAB имеются средства для локального вывода анимированного изображения, эта проблема не всегда решается стандартными средствами даже для локального представления, тем более для представления в Интернете.

### ***Проблемы вывода динамических изображений в локальных задачах и пути их решения***

В настоящее время в системе MATLAB существует два способа получения анимированных изображений — анимация «на лету» с помощью изменения свойств соответствующих графических объектов, и анимация, получаемая с помощью создания соответствующей матрицы с последующим ее выводом с помощью процедуры **movie** [6]. Для создания анимированных демонстраций, обладающих интерактивными возможностями, предпочтительным является первый способ, особенно, в сочетании с графическим интерфейсом, позволяющим оперативно менять параметры задачи и тут же получать динамические результаты расчета на экран. Если задача готовится для самостоятельной работы студента, то предпочтитель-

ным вариантом является анимация «на лету» с одновременной возможностью самостоятельного задания входных параметров. Такая задача должна тщательно разрабатываться и еще с той точки зрения, что необходимо вводить достаточное количество проверок вводимых параметров, поскольку ошибочное (или злонамеренное) введение неверных значений может привести к аварийным ситуациям (деление на ноль, вычисление с несуществующим элементом и т. д.). Если же та же самая задача готовится преподавателем для демонстрации на лекции, то желательно заготовить набор параметров при которых демонстрируемый эффект наиболее ясен, либо приготовить заранее готовые массивы данных с уже записанными анимациями.

При разработке задач по электродинамике часто приходится выводить изолинии скалярного поля или силовые линии (линии тока) векторного поля, причем желательно делать это в режиме анимации если поле меняется со временем. Если для вывода отдельных линий в режиме «анимации на лету» существует подробное описание [4, 6], то для вывода изолиний скалярного поля такого очевидного пути нет. Такая же проблема возникает при динамическом выводе гистограммы. Проблема состоит в том, что процедура вывода изолиний **contour()** и процедура построения гистограммы **hist()** являются процедурами высокого уровня, и они не генерирует дескриптор какого-либо графического объекта, чьи свойства можно было бы менять в процессе вывода для создания эффекта анимации. В этом случае для анимирования гистограммы [4] приходится поступать следующим образом. С помощью процедуры **hist** и **stairs** генерируется (без вывода на экран) набор координат ломаной, которая описывает требуемую гистограмму. Это может быть, например, такой код

```
y= %Генерация случайных чисел
dx=0.25; % Ширина бинов
x=-2.5:dx:2.5; % Центры бинов
% Вычисление параметров гистограммы
[ny,xh]=hist(y,x);
xh=xh-dx/2;
% Вычисление координат ломаной
[xl,yl]=stairs(xh,ny);
% Создание дескриптора ломаной линии
hl=line(xl,yl);
set(hl,'EraseMode','xor');
pause;
```

После этого уже можно запускать вычислительный цикл, в котором изменяются свойства линии **XData** и **YData**, что и приведет к анимированному выводу гистограммы.

Для создания анимированной картины движения изолиний пришлось использовать более сложный алгоритм, хотя суть его осталась такой же — выделение с помощью стандартных функций (в случае с изолиниями это функция **contourc**) векторов координат и дескрипторов линий, которые

описывают искомые изолинии, присвоение им соответствующих свойств 'EraseMode' и постоянное обновление их координат в процессе движения изолиний. Трудность еще состояла в том, что в силу физической специфики задачи и ограниченной области вывода изолиний число изолиний изменялось в процессе вывода. Все это требовало специальных решений при разработке соответствующих процедур. Так например, для решения таких разных задач как изображение силовых линий поля излучения с помощью вектора Герца и изображения силовых линий магнитного поля и токов на стенках прямоугольных волноводов была разработана функция **move\_contour**, которая приводится далее.

```
function [v,hl]=move_contour(x,y,Z,Niz,ini,hl)
% Эта процедура позволяет изобразить движущиеся
% изолинии переменного поля
% x и y - координаты прямоугольной сетки в узлах % которой рассчитано выводимое скалярное поле
% Y — матрица значений скалярного поля в узлах
% Niz — если это целое число, то это число выводимых изолиний, а если это вектор, то его значения — значения поля на выводимых изолиниях
```

```
persistent NNI
% Вычисление матрицы изолиний
CC=contourc(x,y,Z,Niz);
n1=0;
n3=length(CC);
n2=1;
ii=0;
while n2<n3
    ii=ii+1;
    n1=n1+2;
    n2=n1-1+CC(2,n1-1);
    if ini
        % Выделение отдельной линии
        hl(ii)=line(CC(1,n1:n2),CC(2,n1:n2));
        v(ii)=CC(1,n1-1);
        set(hl(ii),'EraseMode','Xor','color','k');
    else
        % Выделение отдельной линии
        set(hl(ii),'Xdata',CC(1,n1:n2),'Ydata',...
            CC(2,n1:n2));
    end;
    n1=n2;
end;% цикла while
if ini
    NNI=length(hl);
elseif NNI>ii
    for iii=ii+1:NNI
        % Удаление «исчезнувших» линий
        set(hl(iii),'Xdata',[],'Ydata',[]);
```

```
end;  
end;
```

### **Вывод анимированных изображений в Интернет**

Ни один из описанных выше способов анимации в чистом виде не годится при представлении задачи в Интернете. Последовательную пересылку и загрузку html-страниц для получения анимации, конечно, реализовать невозможно. Пересылка же матрицы для функции `movie` и ее последующее воспроизведение на стороне клиента требует наличия у клиента установленной системы MATLAB, от чего мы отказались с самого начала. Поэтому нами было принято решение генерировать на стороне сервера средствами MATLAB стандартный для Интернет анимированный графический файл. Система MATLAB до версии 6.0 таких средств в качестве встроенных функций не содержала. Поэтому после проведенного поиска мы остановились на программе **mpgwrite**, взятой нами из архива Math-Works, и пакета программ **makemovie**, разработанного А. Вейманом (A. Weigman, [wiegmann@math.lbl.gov](mailto:wiegmann@math.lbl.gov)).

Программа **mpgwrite** использует матрицу, подготовленную с помощью стандартных средств Matlab для использования с функцией **movie**. Но дело в том, что функция **getframe**, используемая для этих целей, не работает на стороне сервера и команда `M(j)=getframe` возвращает пустую матрицу. Пришлось создавать матрицу `M` с помощью функции **print**.

```
print -djpeg -r72 skin.jpg  
OutImage = imread('skin.jpg');%  
frame(j1).cdata=OutImage;  
frame(j1).colormap=[];
```

После чего использовать матрицу **frame** в качестве аргумента функции **mpgwrite** для создания mpg-файла.

При использовании функции **makemovie** в пакете поставки имеется функция **makeframe**, которая создает и записывает на диск промежуточные файлы с мгновенными изображениями. В этом пакете имеются варианты создания gif-файлов и mpg-файлов, но вариант с созданием mpg-файлов не работает, поэтому нами в настоящее время используются оба пакета один - для создания gif-файлов и второй - mpg-файлов. Необходимость предоставлять пользователю возможность выбора типа генерируемого файла связана с тем, что просмотр этих файлов осуществляется разными средствами по-разному. Gif-файл просматривается на экране средствами браузера и полученная анимация прокручивается заранее заданное число раз. Mpg-файл просматривается с помощью проигрывателя Windows Media (на компьютерах, на которых установлена операционная система Windows) и этим просмотром можно управлять (останавливать, повторять, просматривать по кадрам)

**Опыт разработки и эксплуатации моделирующих программ**

По описанной технологии к настоящему времени разработаны 11 задач по электродинамике: «Диаграмма направленности антенн», «Токи на поверхности кубического резонатора», «Диаграмма направленности излучения релятивистской частицы», «Движение релятивистской частицы в поле сильной электромагнитной волны», «Нестационарный скин-эффект» и «Скин-эффект внутри проводящего полого цилиндра», «Расплывание пакета в среде с дисперсией», «Фурье-преобразование», «Электростатическое поле зарядов», «Излучение вектора Герца». С этими задачами можно познакомиться на сайтах <http://matlab.tutor.net.ru>, <http://www.phys.nsu.ru:8000>.

При эксплуатации задач выяснилась следующая проблема - аварийный останов программы (деление на ноль, несовпадение размерности массивов и т. д.), т. е. те ошибки, которые при локальном исполнении приводят к останову исполнения задачи и переходу в режим командного окна, приводит к зависанию программы `matweb.exe` и необходимости его перезапуска. Кроме того, использованная система генерации графических файлов с записью на диск промежуточных картинок является достаточно сложной и относительно медленной. Выходом может быть разработка функции, генерирующей графические анимированные файлы в памяти. Возможно, появившаяся в версии 6.X функция **avifile** поможет решить эту проблему.

В настоящее время все разработанные задачи могут работать также в локальном варианте, если на компьютера пользователя установлен MatLab. При работе в пределах Новосибирского научного центра время получения результата расчета составляет для разных задач от 30 секунд до 2 минут.

**Литература**

1. Альтшуль Л. М., Мушер С. Л. Применение терминальной вычислительной системы в учебном процессе на физическом факультете Новосибирского университета // Сб.: Автоматизированные системы управления вузом.— Новосибирск: НГУ, 1978.
2. Коткин Г. Л., Фрумин Л. Л. Практикум «Моделирование физических явлений».— Новосибирск: НГУ, 1992.
3. Коткин Г. Л., Черкасский В. С. Численное моделирование физических процессов: Учебное пособие.— Новосибирск: Новосиб. ун-т., 1998.— 124 с.
4. Коткин Г. Л., Черкасский В. С. Численное моделирование физических процессов с использованием MATLAB: Учебное пособие.— Новосибирск: Новосиб. ун-т., 2001.— 173 с.
5. Marchand P. Graphics and GUIs with MATLAB. — 2<sup>nd</sup> ed, CRC press LLC, 1999.

УДК 519.6

## **MATLAB В УЧЕБНЫХ КУРСАХ ПО ИССЛЕДОВАНИЮ ПЕРЕХОДНЫХ ПРОЦЕССОВ В ЭЛЕКТРОЭНЕРГЕТИЧЕСКИХ СИСТЕМАХ**

*Кудашов А. В., Медведева С. Н., Михотин В. Д., Пискарев С. П.  
Пензенский госуниверситет, Пенза,  
e-mail: rgu@penza.net*

Представляемые в настоящем докладе лабораторные работы выполняются с использованием имитационных моделей, реализованных средствами пакета Power System Blockset (PSB), который является эффективным инструментом для моделирования энергетических (силовых) систем и устройств — от простых электрических цепей постоянного и переменного тока до сложных линий электропередачи большой мощности, преобразовательных устройств на современной элементной базе и электрических машин с системами управления [2..4]. Пакет Power System Blockset позволяет создавать PSB-модели электронных и энергетических систем, а также и устройств, характеризующихся объединением электрических цепей с электромеханическими устройствами и различными преобразователями электрической энергии.

При выполнении лабораторных работ студентам предлагаются базовые PSB-модели исследуемых объектов, которые необходимо соответствующим образом адаптировать согласно программам исследований. Для этого предусматривается использование следующих библиотек компонент пакета PSB:

**Connectors** — подключающие и соединительные устройства;  
**Electrical Sources** — источники электрической энергии и сигналов;  
**Elements** — компоненты электрических и электронных устройств;  
**Power Electronics** — устройства энергетической электроники;  
**Machines** — электрические машины;  
**Measurements** — измерительные устройства и блоки управления;  
**Extra Library** — специальные энергетические устройства.

Кроме того, для обработки экспериментальных данных и отображения результатов исследований используются средства пакета Simulink, который так же как и PSB входит в базовую систему MATLAB.

Ниже приводятся описания лабораторных работ по исследованию переходных процессов как в простых электрических цепях, так и в сложных электроэнергетических устройствах, включая воздушные линии связи с распределенными параметрами, трансформаторы и электрические двигатели переменного и постоянного тока.

Представляемые лабораторные работы разработаны и апробированы на кафедре «Автоматизированные электроэнергетические системы» Пензенского госуниверситета и предназначены для студентов специальности 100200 «Электроэнергетические системы и сети» для дневной и заочной форм обучения.

### **Лабораторная работа № 1. Исследование переходных процессов в линейных электрических цепях**

*Цель работы.* Исследование при помощи PSB-моделей переходных процессов в RLC-цепях при включении на постоянное напряжение.

*Описание объекта исследования.*

Используемая для исследования PSB-модель (рис.1.1) находится в файле RLC\_branch, в которой напряжение DC Voltage Source равно 100 В,  $R = 100 \text{ Ом}$ ,  $L = 1 \text{ мГн}$ ,  $C = 1 \text{ мкФ}$ .

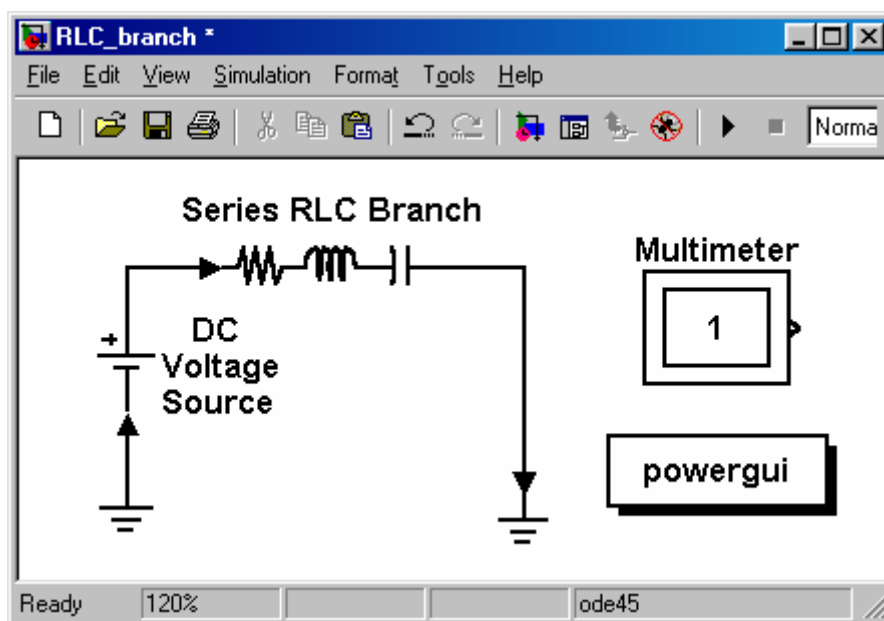


Рис. 1.1. PSB-модель для исследования переходных процессов в последовательных RLC-цепях.

Для измерения тока, протекающего через нагрузку, используется блок Multimeter, окно настройки которого показано на рис. 1.2. Этот блок позволяет измерить электрические переменные, выбранные в окне настройки соответствующих элементов схемы модели. Например, в окне настройки компоненты Series RLC Branch рис. 1.3 имеется возможность заказать измерения целого ряда параметров.



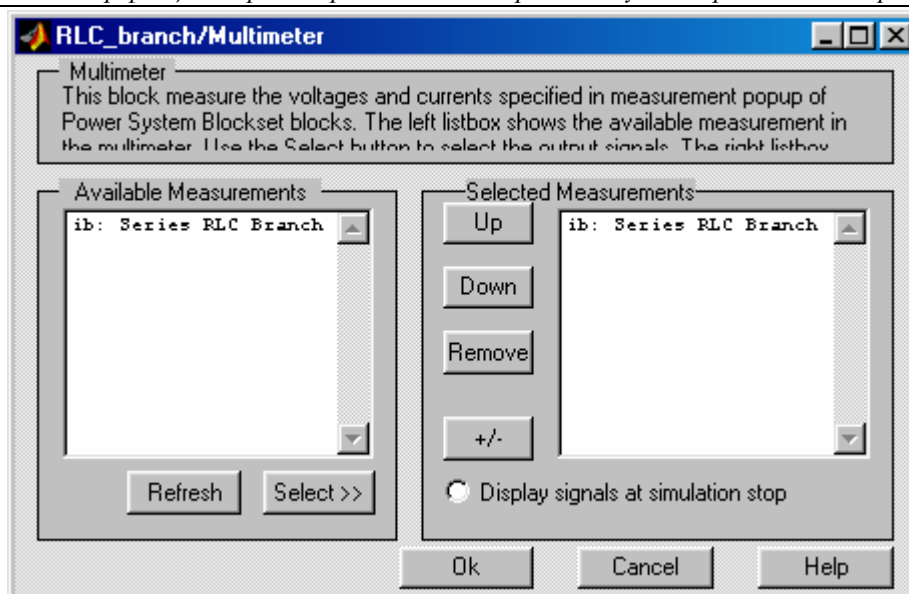


Рис. 1.2. Окно настройки блока Multimeter.

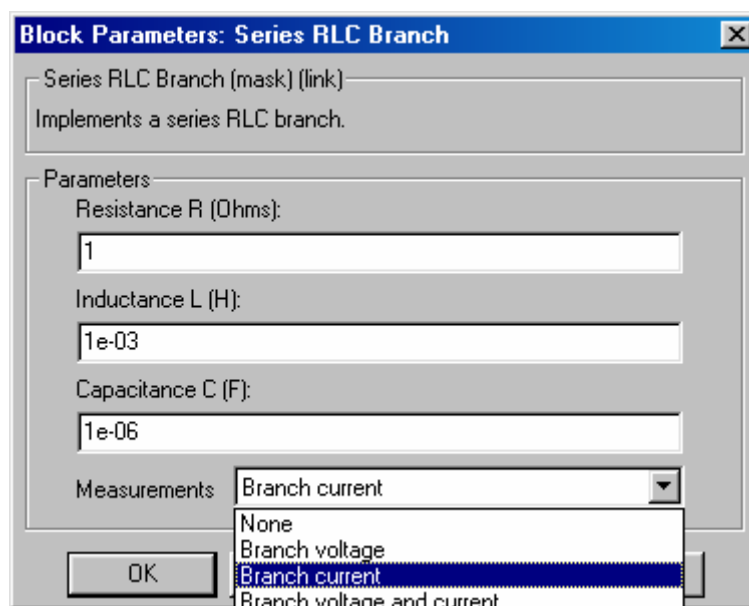


Рис. 1.3. Окно настройки блока Series RLC Branch.

Для отображения переходных процессов в модели используется пользовательский интерфейс powergui.

*Программа исследований.*

1. Ознакомиться с моделью и изучить настройки блоков PSB-модели.
2. Используя исходную модель, исследовать характер переходных процессов в режиме подключения постоянного напряжения. При этом необходимо:

- 2.1. Определить зависимость времени установления от значения сопротивления резистора R.

- 2.2. Построить графики изменения напряжений на L и C.

3. Расчетным путем определить значения  $L$  и  $C$ , при которых наблюдается резонанс напряжений в цепи и, используя PSB-модель построить графики соответствующего переходного процесса для изменения тока в цепи.

*Методические указания по порядку проведения работы.*

1. Для исследования переходных процессов используется блок `powergui`. При этом выполняется следующая последовательность действий или активизация и подключение мышью предлагаемых функций меню: `powergui` → Tools → Control System Toolbox (LTI Viewer) → Open new LTI Viewer. После выполнения последней операции на экране появляется график рис. 1.4. переходного процесса, отображающий реакцию цепи на скачкообразное изменение входного напряжения

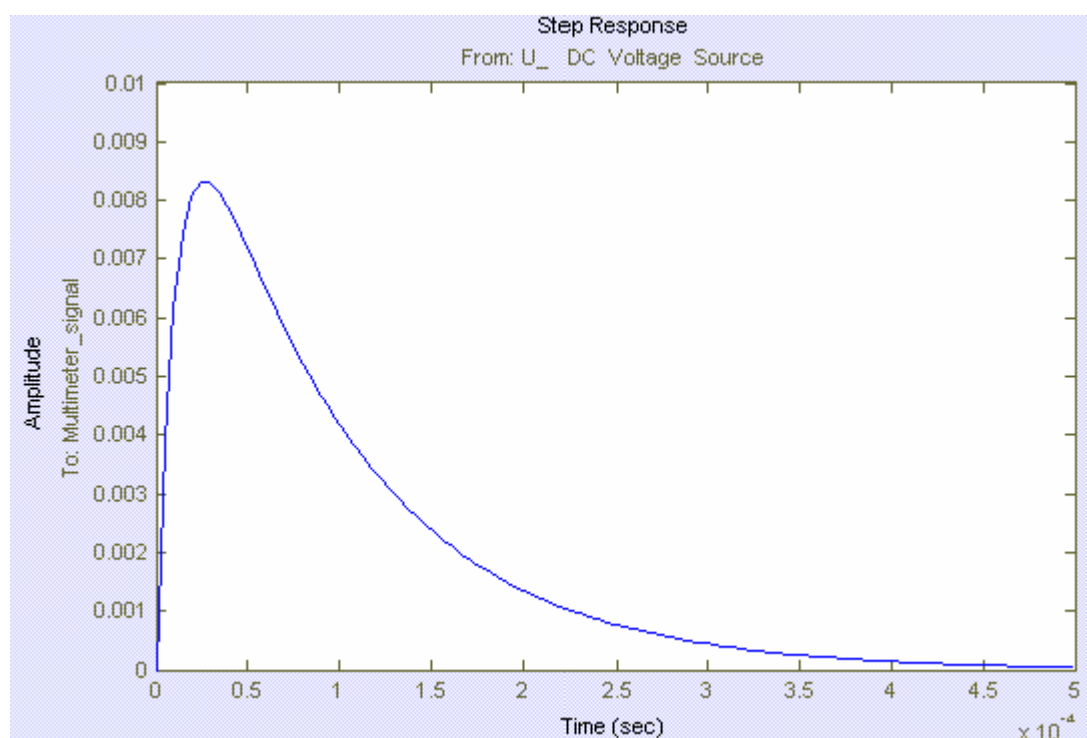


Рис. 1.4. График переходного процесса.

## **Лабораторная работа № 2. Исследование переходных процессов в параллельном LC контуре**

*Цель работы.* Исследование при помощи имитационных PSB-моделей переходных процессов в параллельном LC контуре.

*Описание объекта исследования.*

Используемая для исследования PSB-модель (рис.2.1) находится в файле **LC\_branch**. В представленной модели заданы следующие исходные параметры:  $U_0 = 125$  В,  $R = 100$  Ом,  $L = 40$  мГн,  $C = 5$  мкФ.

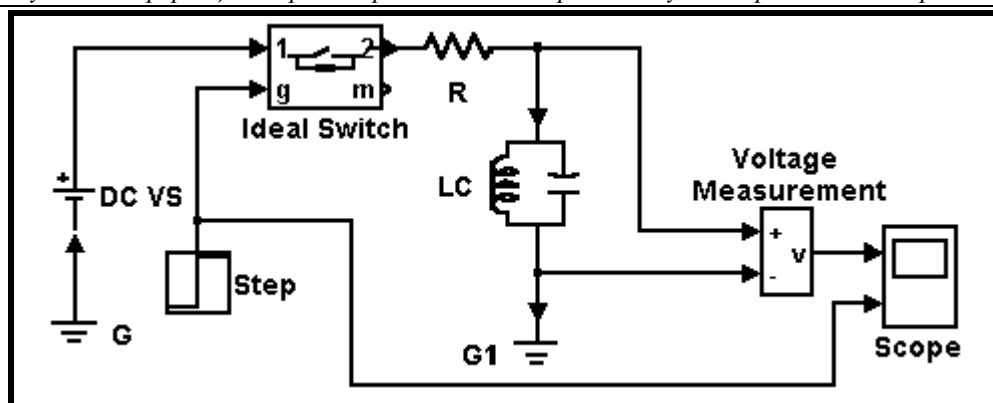


Рис. 2.1. Базовая PSB-модель исследуемой схемы.

График изменения напряжения на контуре показан на рис. 2.2.

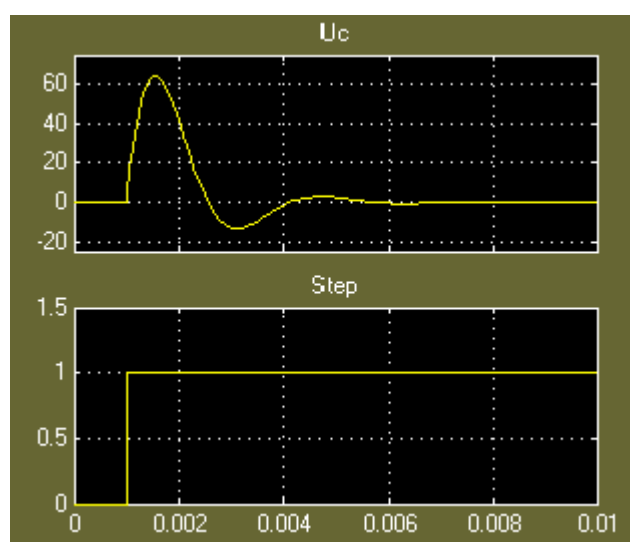


Рис. 2.2. Контрольные осциллограммы.

### Программа исследований.

1. Ознакомиться с моделью и изучить настройки блоков PSB-модели.
2. Используя исходную модель, исследовать характер переходных процессов в режиме подключения постоянного напряжения. При этом необходимо:
  - 2.1. Определить зависимость времени установления от значения сопротивления балластного резистора.
  - 2.2. Определить значение резонансной частоты и добротность контура.
  - 2.3. Построить графики изменения токов через  $L$  и  $C$ .
3. Заменить источник постоянного напряжения на источник переменного напряжения с параметрами  $U_m = 200$  В и  $f = 50$  Гц. Провести исследование переходных процессов при различных значениях начальной фазы входного напряжения.

4. Исключить из PSB-модели ключ Ideal Switch и экспериментально определить значение резонансной частоты.

*Методические указания по порядку проведения работы.*

1. Длительность установления переходного процесса обычно определяется как интервал времени от начала входного воздействия на цепь до момента затухания исследуемого сигнала с погрешностью 1%. Для измерения максимального значения исследуемого сигнала и момента окончания переходного процесса в системе Power System Blockset удобно использовать кнопки Zoom Y-axis и Zoom X-axis, которые находятся на панели управления осциллографов (Scope).

2. Для измерения частоты резонанса при  $R = 500$  Ом следует воспользоваться двумя методами.

Первый метод предусматривает измерение периода колебаний напряжения на контуре и вычисление по известным формулам резонансной частоты.

Второй метод — непосредственного подбора частоты резонанса — требует собрать PSB-модель показанную на рис. 2.3, которая содержит необходимые датчики тока и напряжения, а также блок измерения токов и напряжений **powergui**. Значению резонансной частоты соответствует совпадение амплитуд токов  $I_L$  и  $I_C$ .

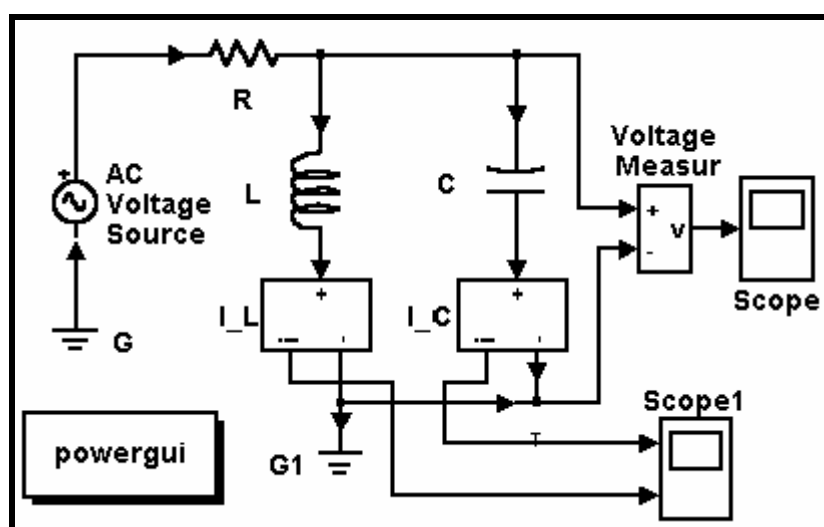


Рис.2.3. PSB-модель исследуемой схемы для измерения резонансной частоты.

Для проверки результатов сравните полученные данные с теоретическими, которые вычисляются по формуле [1]:

$$f_p = \frac{1}{2\pi\sqrt{LC}}.$$

3. Теоретически добротность контура при резонансе токов определяется по формуле:

$$Q = \frac{R}{\sqrt{L/C}}.$$

Практически добротность можно оценить по числу колебаний совершаемых переходной характеристикой рис 2.2. от начала процесса до момента его установления.

### **Лабораторная работа № 3. Исследование переходных процессов в линиях электропередачи с распределенными параметрами**

*Цель работы.* Исследование при помощи имитационных PSB-моделей переходных процессов при включении и выключении длинных воздушных линий с распределенными параметрами.

*Описание объекта исследования.*

Используемая для исследования PSB-модель (рис.3.1) находится в файле **LINE\_DPS\_1**.

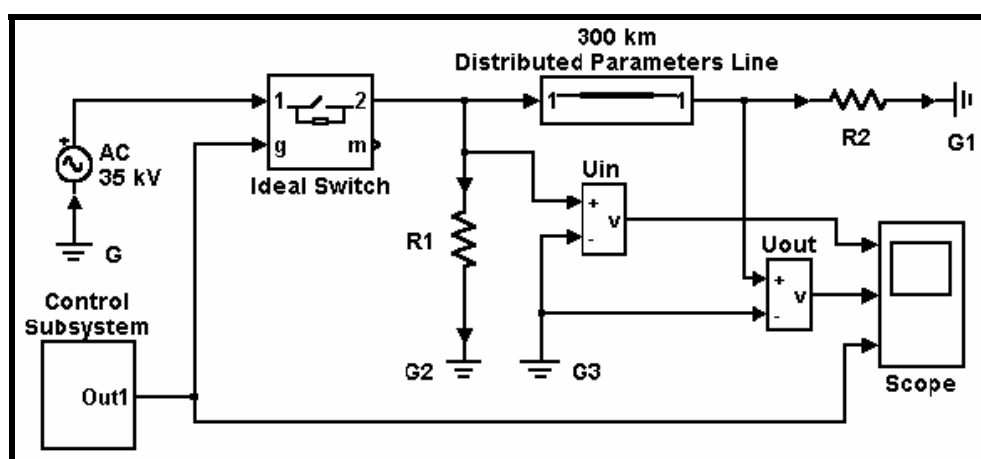


Рис. 3.1. PSB-модель для исследования линии с распределенными параметрами.

В модель кроме компонент, значение которых понятно из рис. 3.1, входит в качестве основного блока компонента Distributed Parameters Line — модель воздушной линии электропередачи с распределенными параметрами, окно настройки которой показано на рис. 3.2. Для имитации процессов включения и выключения линии в схеме используется специальный блок управления Control Subsystem.

Parameters

Number of phases N  
1

Frequency used for R L C specification (Hz)  
50

Resistance per unit length (Ohms/km) [N\*N matrix] or [R1 R0]  
0.2568

Inductance per unit length (H/km) [N\*N matrix] or [L1 L0 L0m]  
2.0e-3

Capacitance per unit length (F/km) [N\*N matrix] or [C1 C0 C0m]  
8.6e-9

Line length (km)  
300

Measurements Phase-to-ground voltages

Рис. 3.2. Окно настройки блока Distributed Parameters Line.

Графики изменения входного  $U_{in}$  и выходного  $U_{out}$  напряжений, а также управляющее ключом воздействие показаны на рис. 3.3. Представленные результаты получены при работе линии в режиме холостого хода ( $R_1 = 50$  кОм,  $R_2 = 1$  МОм).

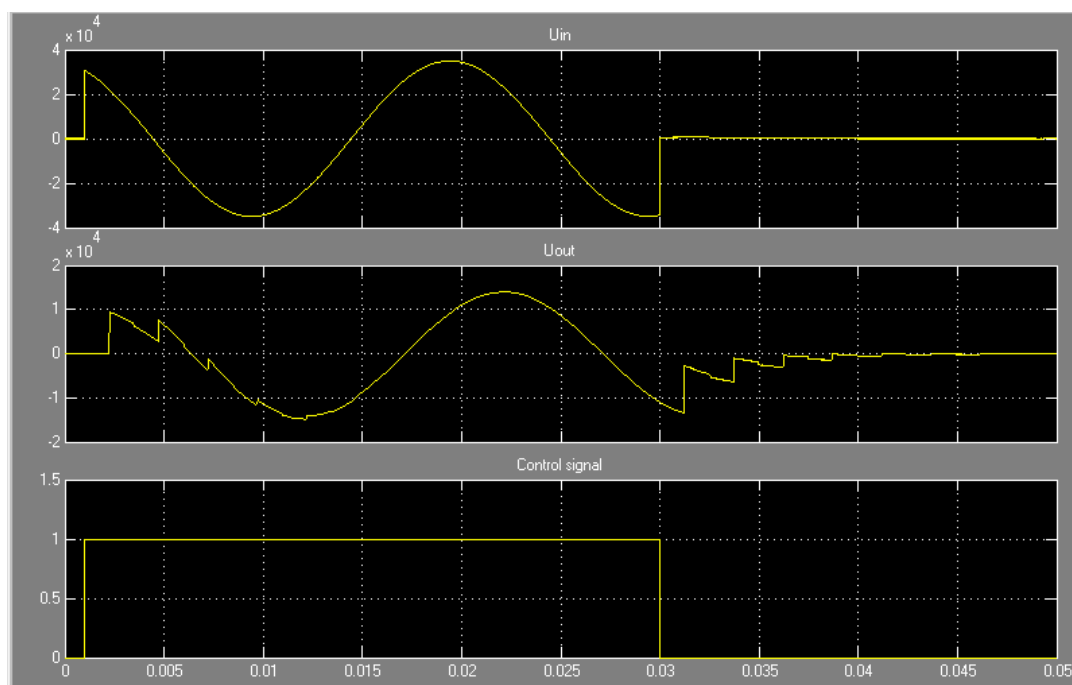


Рис. 3.3. Контрольные результаты моделирования линии с распределенными параметрами.

### *Программа исследований.*

1. Используя исходную модель рис. 3.1, исследовать характер переходных процессов в режиме холостого хода (без нагрузки) и в режимах максимальной нагрузки. При этом необходимо:

1.1. Определить зависимость времени задержки (запаздывания) включения от размера нагрузки.

1.2. Определить значение волнового сопротивления линии и размер согласованной нагрузки.

1.3. Определить значение скорости распространения электромагнитной волны по линии.

2. Построить PSB-модель и провести необходимые исследования переходных процессов по активной и реактивной мощностям. При этом необходимо:

2.1. Исследовать характер переходного процесса в зависимости от размера и типа нагрузки (активной, индуктивной и емкостной).

2.2. Исследовать характер переходного процесса при согласованной нагрузке.

2.3. По результатам моделирования оценить КПД исследуемой линии.

### *Методические указания по порядку проведения работы.*

1. При оценке значения волнового сопротивления  $Z_{\text{в}}$  следует учитывать, что если  $R_2 \neq Z_{\text{в}}$ , то падающая волна частично пройдет в нагрузку и частично отразится от нее (возникает отраженная волна) [1]. Обычно берут  $R_2 = Z_{\text{в}}$ . Такую нагрузку называют согласованной, при ней отраженная волна отсутствует. Исходя из этого, необходимо экспериментально подобрать значение  $R_2$ , при котором отсутствует отраженная волна.

2. Для исследования переходных процессов по активной и реактивной мощностям необходимо собрать PSB-модель показанную на рис. 3.4. В данной модели резистор **R2** исходной модели рис. 3.1 заменяется элементом нагрузки Parallel RLC Load, при настройке которого необходимо указать рабочие номинальное напряжение линии и частоту. Дополнительно в модель вводятся датчики тока Iin, Iout (PSB компоненты Current Measurement) и измерители активной и реактивной мощностей Active & Reactive Power, которые находятся в библиотеке Extra Library + Measurements. Использование блока Active & Reactive Power предусматривает его настройку по частоте.

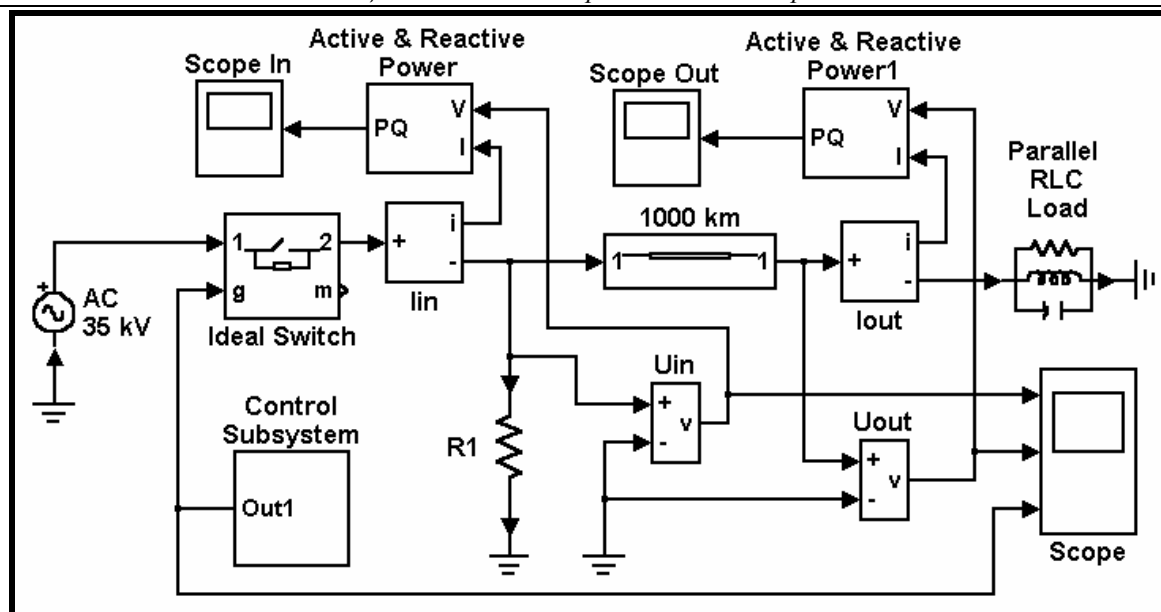


Рис. 3.4. PSB-модель для исследования линии с распределенными параметрами.

3. При необходимости изменения времени управляющих ключом воздействий необходимо произвести перенастройку блоков Step входящих в подсистему Control Subsystem (см. рис. 3.5).

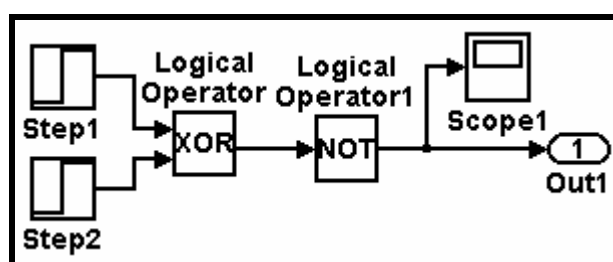


Рис. 3.5. PSB-модель подсистемы Control Subsystem.

#### **Лабораторная работа № 4 Исследование переходных процессов в линейных трансформаторах.**

**Цель работы.** Исследование при помощи имитационных PSB-моделей переходных процессов при включении линейного трансформатора-преобразователя однофазного напряжения в трехфазное.

**Описание объекта исследования.**

Используемая для исследования PSB-модель (рис. 4.1) находится в файле **LT\_model**. Исследуемый Linear Transfomer имеет настройки, которые показаны на рис. 4.2.

Среди компонент PSB-модели, назначение которых понятно из пояснений к предыдущим лабораторным работам, присутствует блок



Breaker — прерыватель имитирующий обрыв нагрузки в одной из фаз (см. рис. 4.3).

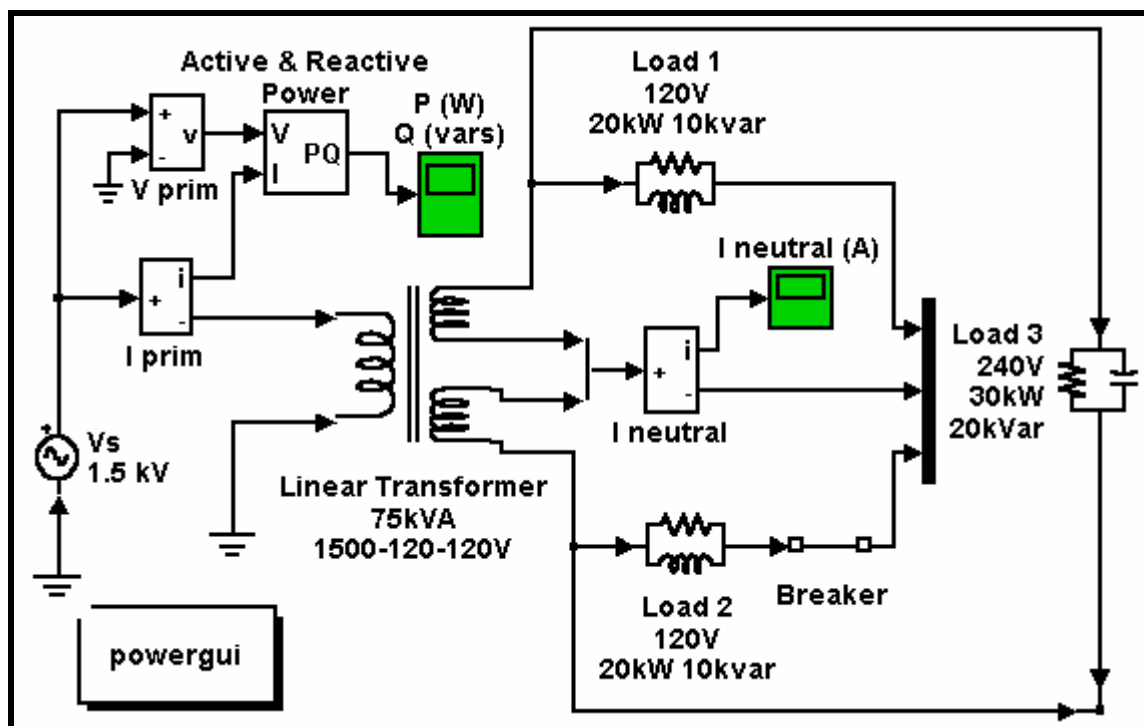


Рис. 4.1. Модель линейного трансформатора

Parameters

Nominal power and frequency [Pn(VA) fn(Hz)]:

Winding 1 parameters [V1(Vrms) R1(pu) L1(pu)]:

Winding 2 parameters [V2(Vrms) R2(pu) L2(pu)]:

Winding 3 parameters [V3(Vrms) R3(pu) L3(pu)]:

Magnetization resistance and reactance [Rm(pu) Lm(pu)]:

Measurements

Рис. 4.2. Окно настройки модели линейного трансформатора.

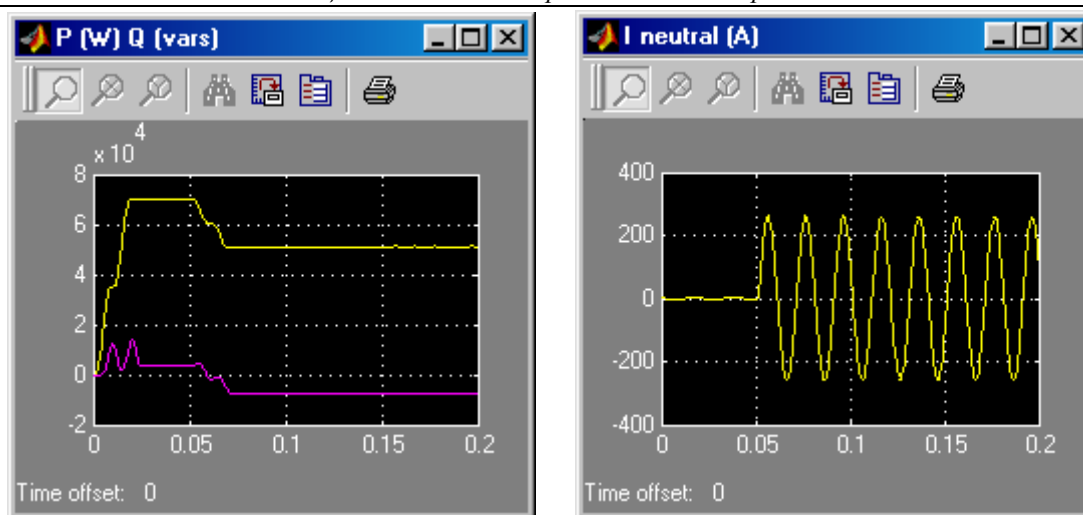


Рис. 4.3. Контрольные осциллограммы процессов.

*Программа исследований.*

1. Ознакомиться с моделью и изучить настройки блоков PSB-модели.  
 2. Используя исходную модель, исследовать характер переходных процессов в режиме размыкания прерывателя Breaker. При этом необходимо:

2.1. Определить при начально-заданных нагрузках напряжения, токи и мощности в цепях схемы, а также время установления переходного процесса.

2.2. Повторить эксперименты п.п. 4.1, но при индуктивных и емкостных нагрузках.

3. Исследовать переходные процессы в режиме включения трансформатора при номинальной нагрузке и в режиме холостого хода. При этом отследить, как переходные процессы зависят от фазы питающего напряжения в момент включения.

4. Повторить эксперименты п.п. 3 при различных видах нагрузки: активной, индуктивной и емкостной.

*Методические указания по порядку проведения работы.*

1. Для выполнения п.п. 2 программы исследований в исходную PSB-модель дополнительно требуется ввести необходимые датчики тока Current Measurement и напряжения Voltage Measurement, а также средства отображения получаемой информации Scope.

2. При выполнении п.п. 3 и 4 программы исследований в исходной PSB-модели исключается прерыватель Breaker и дополнительно вводится ключ, который имитирует процесс включения первичной обмотки трансформатора. Необходимый ключ и управляющий контроллер можно скопировать из PSB-модели рис. 2.1 применяемой в лабораторной работе №2 (см. рис. 4.4.)

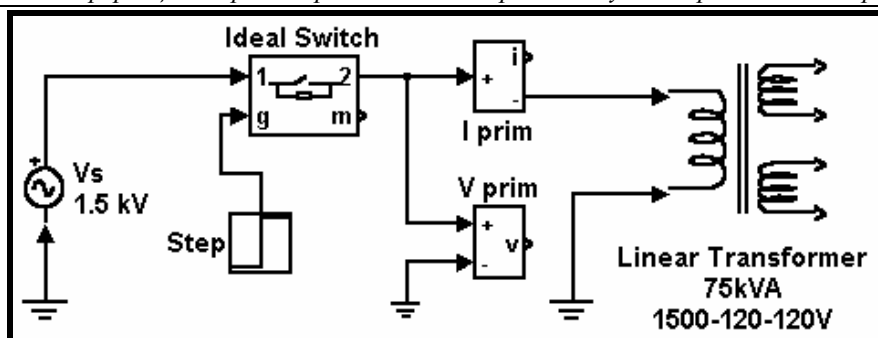


Рис. 4.4. Схема подключения трансформатора.

3. При выполнении лабораторной работы значения нагрузок задавать после предварительных консультаций с преподавателем.

### **Лабораторная работа № 5. Исследование переходных процессов в трехфазных асинхронных электрических машинах**

*Цель работы.* Исследование при помощи имитационных PSB-моделей переходных процессов при пуске трехфазных асинхронных электрических машин.

*Описание объекта исследования.*

Используемая для исследования PSB-модель (рис. 5.1) находится в файле **AS\_Mashin1**.

В состав модели входят следующие основные блоки/модели из состава библиотек Power System Blockset.

**Inductive source with neutral** — трехфазный генератор переменного напряжения, PSB-модель которого показана на рис. 5.2. Используемые в работе настройки генератора приведены на рис. 5.3.

**Tree-Phase V-I Measurement** — блок датчиков тока и напряжения для измерений во входных цепях асинхронной машины. Исходная (входящая в состав библиотеки Extra Library\Measurements) схема подключения датчиков приведена на рис. 5.4.

Для удобства работы и вывода результатов непосредственно в значениях мгновенных токов и напряжений в данном блоке следует исключить компоненты **Ki** и **Kv**, с помощью которых осуществляется нормирование, и, кроме того, в модель полезно ввести панель вывода данных **powergui**. Преобразованный подобным образом блок Tree-Phase V-I Measurement не требует каких-либо настроек.

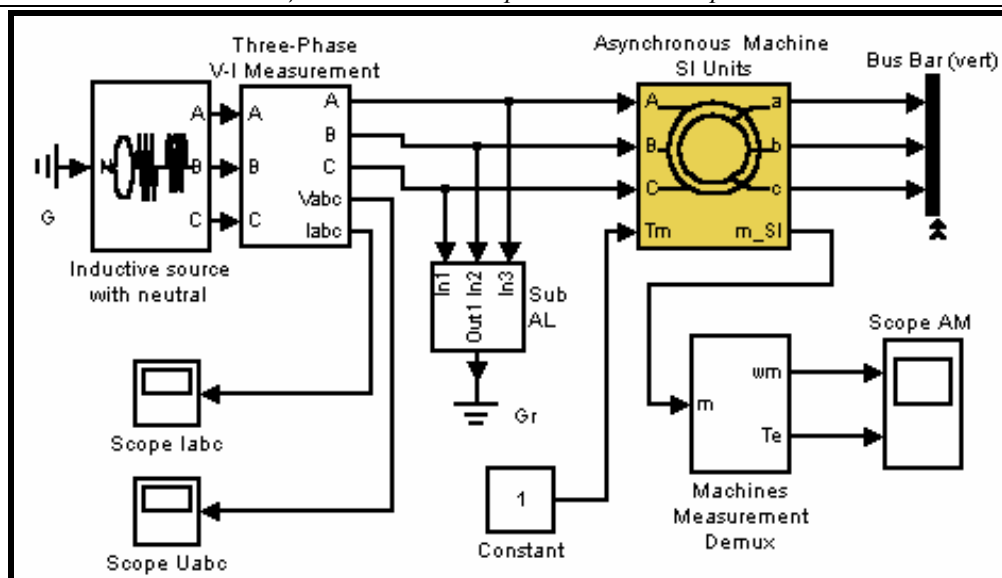


Рис. 5.1. PSB-модель асинхронной машины.

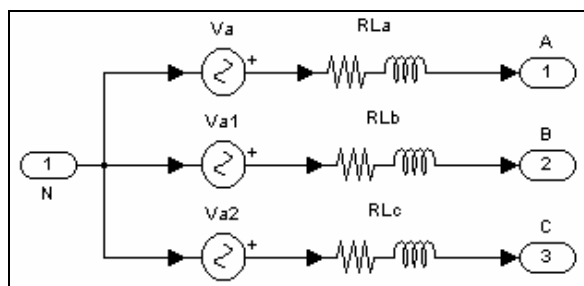


Рис. 5.2. PSB-модель блока Inductive source with neutral.

Parameters

Phase-to-ground peak voltage (V) :

Phase angle of phase A (Degrees) :

Frequency (Hz) :

Source resistance (Ohms):

Source inductance (H) :

Рис. 5.3. Окно настроек блока Inductive source with neutral.

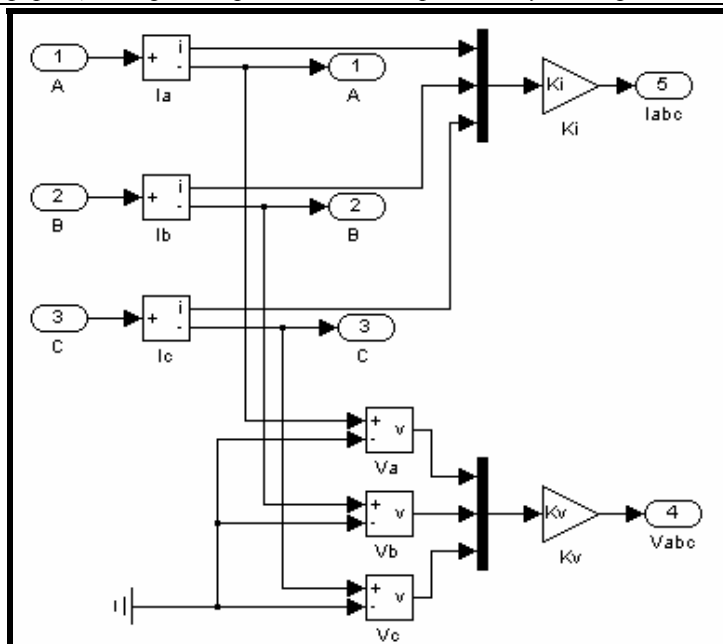


Рис. 5.4. Исходная PSB-модель блока Tree-Phase V-I Measurement.

**Asynchronous Machine SI Units** — модель трехфазной асинхронной машины, обмотки статора которой обозначены буквами ABC, а ротора — abc. Настройка параметров модели выполняется согласно рис. 5.5. Для задания нагрузочного момента к входу  $T_m$  подключен блок Constant.

Parameters:

Rotor type:

Reference frame:

Nom. power, L-L volt. and freq. [ Pn(WA), Vn(Vrms), fn(Hz) ]:

Stator [ Rs(ohm) Lls(H) ]:

Rotor [ Rr(ohm) Llr(H) ]:

Mutual inductance Lm (H):

Inertia, friction factor and pairs of poles [ J(kg.m^2) F(N.m.s) p ]:

Initial conditions [ s() th(deg) isa, isb, isc(A) pha, phb, phc(deg) ]:

Рис. 5.5. Окно настроек блока Asynchronous Machine SI Units.

**Mechines Measurement Demux** — селектор измеряемых параметров исследуемой асинхронной машины. Выбор необходимых параметров осуществляется из меню, представленного на рис. 5.6.

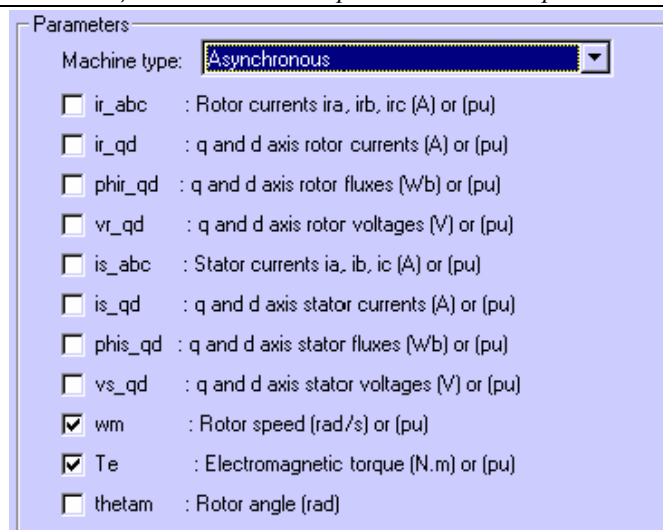


Рис. 5.6. Окно настройки блока измерений параметров машины Mechines Measurement Demux.

Кроме того, модель содержит стандартные схемы отображения информации из библиотеки Simulink и подсистему Sub AL, которая имитирует потери в соединительных проводах. Включение этой подсистемы необходимо для корректной работы модели. Правильность настройки исходной модели можно проконтролировать по временным диаграммам на осциллографе Scope AM, которые показаны на рис. 5.7.

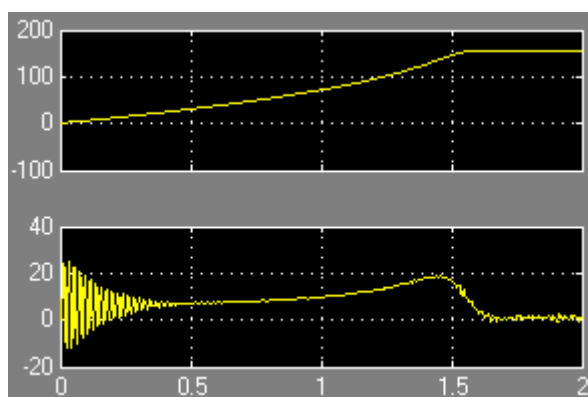


Рис. 5.7. Контрольные осциллограммы работы асинхронной машины.

### *Программа исследований.*

1. Используя исходную модель, исследовать характер переходных процессов в режиме холостого хода (без нагрузки) и в режиме максимальной нагрузки. При этом необходимо:

1.1. Определить зависимость времени разгона двигателя от значения вращающего момента на валу.

1.2. Определить зависимость установившейся скорости вращения от значения вращающего момента на валу.

1.3. Построить графики изменения мгновенных и действующих значений токов и напряжений в обмотках статора асинхронной машины при различных нагрузках в режиме пуска.

1.4. Построить графики изменения активной и реактивной мощности в обмотках статора асинхронной машины при различных нагрузках в режиме пуска.

2. Исследовать характер переходных процессов в асинхронной машине при различных значениях выходных параметров трехфазного генератора, посредством которых имитируется мощность сети питания. Соответствующие исследования провести для параметров и характеристик указанных в п.п. 1.1.–1.4.

3. Используя возможности подсистемы Sub AL для имитации компенсирующих конденсаторных батарей, исследовать характер переходных процессов в асинхронной машине при различных значениях емкости компенсирующих конденсаторов. При этом исследования провести для параметров и характеристик указанных в п.п. 1.1.–1.4.

*Методические указания по порядку проведения работы.*

1. Для измерения скорости вращения выходного вала в об/мин к исходной модели добавить блоки нормирования (Gain) и цифрового измерения (Display), которые можно взять из библиотек Simulink. Пример подключения блоков показан на рис. 5.8.

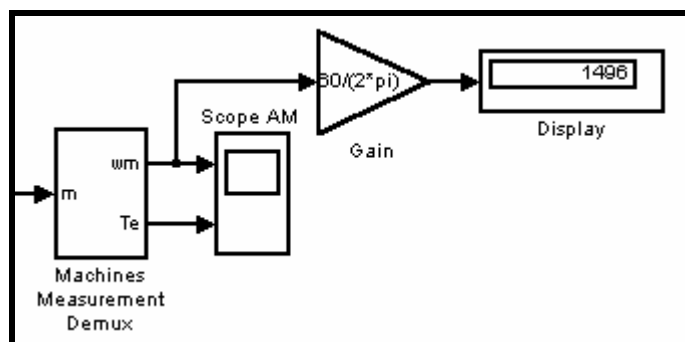


Рис. 5.8. Блоки для измерения скорости вращения вала в об/мин.

2. Для измерения в переходном режиме действующих значений напряжений и токов в обмотках статора в исходной модели сделайте следующие дополнения.

К выходам блока Tree-Phase V-I Measurement подключите демультиплексоры, которые имеются в библиотеке Simulink \ Signals & Systems. Пример демультиплексирования вектора  $I_{abc}$  показан на рис. 5.9.

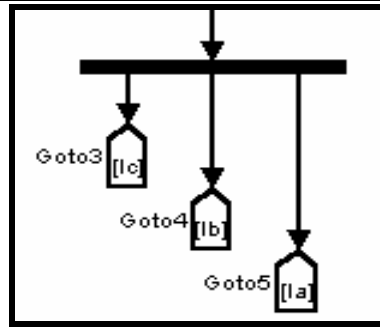


Рис. 5.9. Демультимплексор вектора данных Iabc.

Используя информацию с выходов компонент Goto и блоки измерения действующих значений (RMS) из библиотеки Extra Library\Measurements, организуйте подсистемы для измерения соответствующих токов и напряжений. В качестве примера на рис. 5.10 показана схема для измерения действующих значений фазных токов. На рис. 5.11. приведен характерный пример изменения токов в момент пуска асинхронной машины.

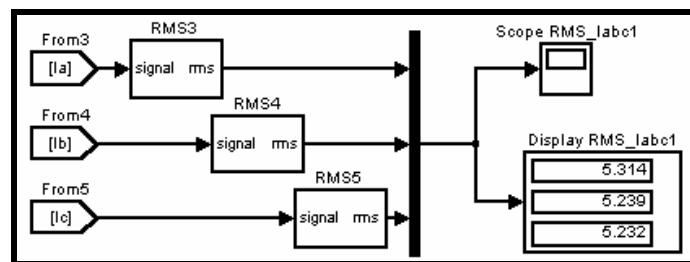


Рис. 5.10. Блок-диаграмма модели для измерения временных зависимостей действующих значений токов Iabc.

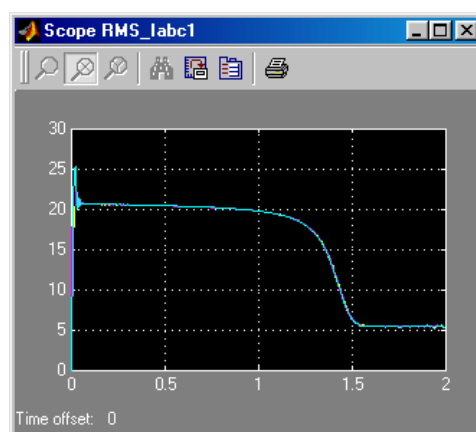


Рис. 5.11. Осциллограммы действующих значений токов Iabc.

3. Для измерения в переходном режиме активной и реактивной мощностей используйте информацию с выходов компонент Goto и блоки измерения Active & Reactive Power из библиотеки Extra Library \ Measurements.



Пример организации подсистемы для измерения суммарных активных и реактивных мощностей показан на рис. 5.12, а характерные кривые изменения  $P$  и  $Q$  приведены на рис. 5.13.

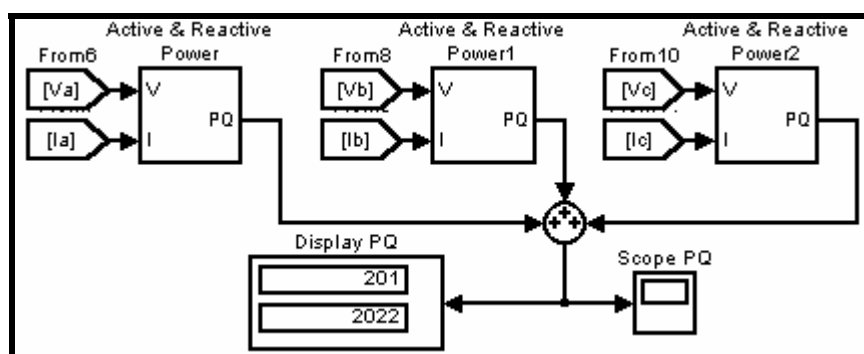


Рис. 5.12. Подсистема для измерения активной и реактивной мощностей.

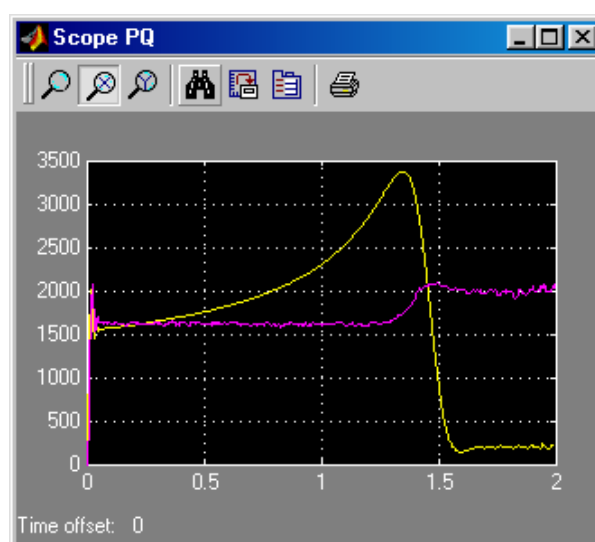


Рис. 5.13. Графики изменения активной и реактивной мощностей.

## Лабораторная работа № 6. Исследование переходных процессов в электромоторах постоянного тока

**Цель работы.** Исследование переходных процессов при пуске электромоторов постоянного тока.

**Описание объекта исследования**

Используемая для исследования PSB-модель (рис.6.1) находится в файле **DC\_motor**.

Объект исследования — электрический мотор постоянного тока DC\_motor моделируется посредством подсистемы, содержание которой показано на рис. 6.2, а соответствующее окно настройки модели на рис. 6.3. В используемой модели входы A+ и A- являются входами коллектора ротора, входы F+ и F- моделируют статор, вход TL воспринимает нагру-

зочный момент на валу, а выход  $m$  в виде четырехмерного вектора обеспечивает вывод информации о параметрах работы мотора: частоте вращения вала  $w$ , токе ротора  $I_a$ , токе статора  $I_f$  и моменте вращения на валу  $T_e$ . Как видно из рис. 6.2, рассматриваемая модель учитывает индуктивности и сопротивления потерь в обмотках ротора и статора.

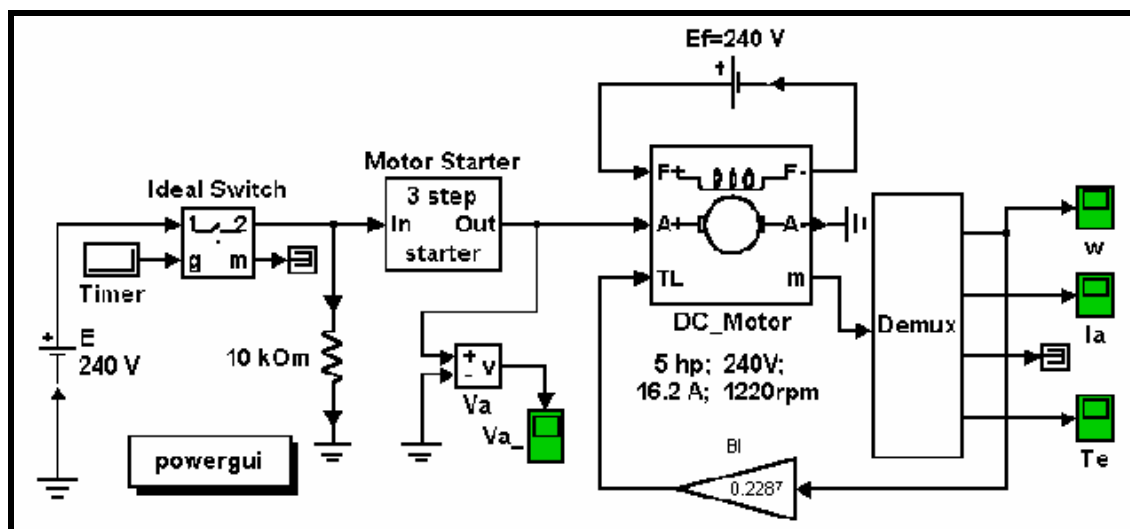


Рис. 6.1. Модель электродвигателя постоянного тока.

Указанные на рис. 6.3 параметры настройки моделируют мотор мощностью 5 л.с. с номинальной скоростью вращения 1220 об/мин.

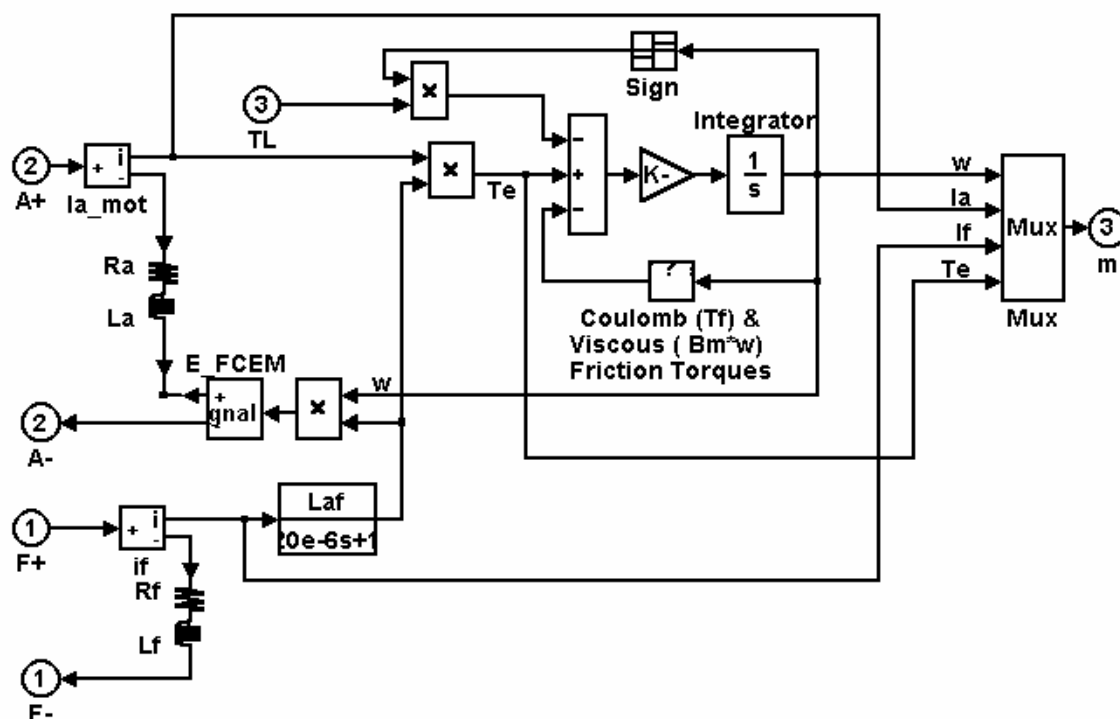


Рис. 6.2. Подсистема DC\_motor.

Процесс «включения» мотора осуществляется блоком Ideal Switch с управляющим таймером и подсистемой Motor Starter, состав которой показан на рис. 6.4. Таким образом имитируется типовая схема старта электродвигателя постоянного тока, которая предусматривает включение в начальный момент в цепь ротора ряда балластных сопротивлений и их отключение в процессе увеличения оборотов вала двигателя.

Parameters

Armature resistance and inductance [Ra (ohms) La (H) ]  
[ 0.6 0.012 ]

Field resistance and inductance [Rf (ohms) Lf (H) ]  
[ 240 120 ]

Field-armature mutual inductance Laf (H) :  
1.8

Total inertia J (kg.m<sup>2</sup>)  
1

Viscous friction coefficient Bm (N.m.s)  
0

Coulomb friction torque Tf (N.m)  
0

Initial speed (rad/s) :  
1

Рис. 6.3. Окно настройки модели DC\_motor.

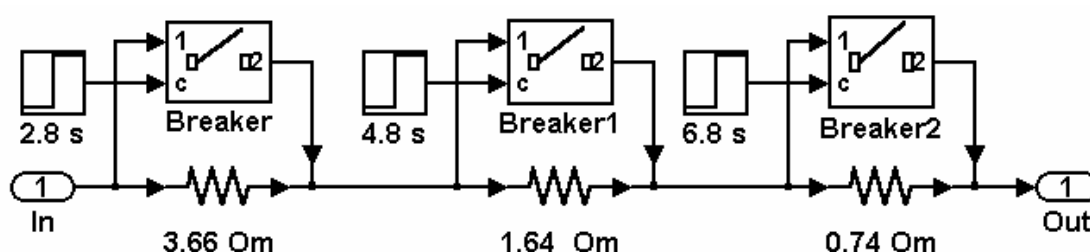


Рис. 6.4. PSB-модель подсистемы Motor Starter.

В рассматриваемой модели нагрузочный момент на валу моделируется как зависящий от скорости вращения вала. Контрольные осциллограммы работы PSB-модели приведены на рис. 6.5 и рис. 6.6.

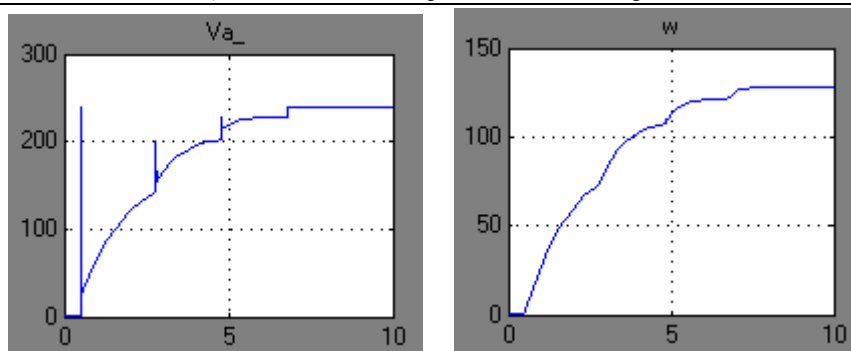


Рис. 6.5. Графики переходных процессов.

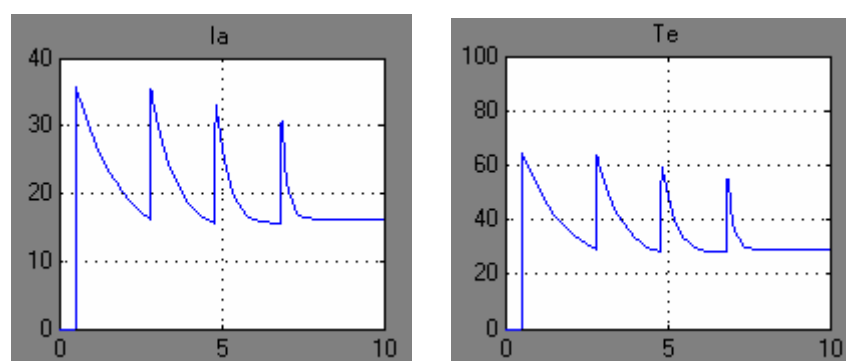


Рис. 6.6. Графики переходных процессов.

*Программа исследований.*

1. Используя исходную модель, исследовать характер переходных процессов в режиме холостого хода (без нагрузки) и в режиме максимальной нагрузки. При этом необходимо:

1.1. Определить зависимость времени разгона двигателя от значения вращающего момента на валу.

1.2. Определить зависимость установившейся скорости вращения от значения вращающего момента на валу.

1.3. Построить графики изменения мгновенных и действующих значений токов и напряжений в обмотках при различных нагрузках в режиме пуска.

1.4. Построить графики изменения активной и реактивной мощности в обмотках при различных нагрузках в режиме пуска.

2. Исследовать характер переходных процессов в машине постоянного тока при различных значениях мощности сети питания.

3. Исследовать характер переходных процессов в машине постоянного тока при различных значениях нагружающего момента на валу двигателя.

*Методические указания по порядку проведения работы.*

1. Для проведения измерений характеристик переходных процессов согласно п. 1 необходимо из исходной модели исключить блок В1 и, ис-

пользуя блок Constant из Simulink-библиотеки Sources, задавать необходимые значения момента TL на валу двигателя.

2. Все необходимые измерения параметров и характеристик выполнять используя методические указания к предыдущим работам.

### ***Литература***

1. Бессонов Л. А. Теоретические основы электротехники.— М.: Высшая школа, 1978.— 752 с.
2. Герман-Галкин С. Г. Компьютерное моделирование полупроводниковых систем в MATLAB 6.0: Учебное пособие. — СПб.: КОРОНА принт, 2001.— 320 с.
3. Гультияев А. К. MATLAB 5.3 Имитационное моделирование в среде Windows: Практическое пособие.— СПб.: Корона принт, 2001.— 400 с.
4. Дьяконов В. П., Круглов В. В. MATLAB. Анализ, идентификация и моделирование систем.— СПб.: Пит, 2002.— 448 с.
5. Дьяконов В. П., Абраменкова И. В., Круглов В. В. MATLAB 5.3.1 с пакетами расширений.— М.: Нолидж, 2001.— 432 с.
6. Петров Г. Н. Электрические машины. В 3-х частях. Ч.1. Введение. Трансформаторы. Учебник для вузов.— М.: Энергия, 1974.— 240 с.

УДК 519.6

## **АНИМАЦИОННАЯ ВИЗУАЛИЗАЦИЯ РЕЗУЛЬТАТОВ МОДЕЛИРОВАНИЯ УПРАВЛЕНИЯ ПОЛЕТОМ В СРЕДЕ MATLAB**

*Мулин П. В., Костиков П. А.  
Московский авиационный институт  
(государственный технический университет), Москва,  
e-mail:mulin@newmail.ru*

При изучении дисциплин «Динамика полета», «Системы автоматического управления летательных аппаратов», «Проектирование систем управления летательных аппаратов» и близких к ним по содержанию важную роль в правильном понимании студентами явлений, свойственных полету самолета, физической сущности эксплуатационных требований, в анализе динамики самолета с системой управления играет имитационное моделирование. Полученные в ходе моделирования результаты необходимо представить в форме, максимально удобной для восприятия и интуитивно понятной студентам. В этой связи интерес представляет применение средств трехмерной визуализации движения самолета, включающие построение 3-D траекторий в процессе моделирования, визуализацию пространственного положения самолета, имитаторов пилотажных приборов. Эту задачу могут решить входящие в MATLAB инструменты визуализации результатов моделирования на экранах набора виртуальных пилотажных приборов (пакет Dials & Gauges Blockset) и 3D анимацию (пакет Virtual Reality Toolbox), которые управляются из среды Simulink. Использование языка моделирования виртуальной реальности VRML позволяет осуществлять анимацию результатов моделирования по ключевым кадрам трехмерной сцены, создаваемым в редакторе V-Realm Builder.

Авторами с использованием редактора V-Realm Builder созданы трехмерные сцены, используемые для анимации результатов моделирования взлета и посадки палубного самолета. Предполагается, что в ходе лабораторного практикума студенты создают Simulink-модель динамики самолета с системой управления, палубы корабля, кинематических связей, подключают блоки пакетов Dials & Gauges Blockset, Virtual Reality Toolbox, Aerospace Blockset, загружают готовую трехмерную сцену для анимации и проводят имитационное моделирование в реальном масштабе времени для заданных начальных условий полета и возмущений. Интерфейс средства отображения виртуальной реальности пакета Virtual Reality Toolbox предоставляет возможность выбора из пяти ракурсов для наблюдения за пространственным движением самолета и позволяет менять положение наблюдателя в подвижной и неподвижной системах отчета.

Предполагается, что данному лабораторному практикуму будут предшествовать занятия в виде лекционного и лабораторного практикума по работе в среде MATLAB в рамках дисциплин: «Вычислительные алгоритмы» и «Системы автоматизированного проектирования систем управления».

Средства трехмерной визуализация движения самолета упрощают отладку моделей, увеличивают наглядность моделирования, облегчают физическую трактовку функционирования системы управления, активизируют учебно-познавательную деятельность студентов.

### ***Литература***

1. MATLAB Documentation.— The Mathworks Inc., 2004.  
(<http://www.mathworks.com>).

УДК 550.383.2

## **ПРИМЕНЕНИЕ MATLAB WEB SERVER В ГЕОФИЗИКЕ ДЛЯ ИНТЕРАКТИВНОЙ АДАПТИВНОЙ ОБРАБОТКИ ДАННЫХ, РАСПРЕДЕЛЕННЫХ В СЕТИ ИНТЕРНЕТ**

*Одинцов В. И.,*

*Институт Земного магнетизма, ионосферы и  
распространения радиоволн РАН, Троицк, Московская обл.,  
e-mail: vodin@izmiran.ru*

*Конрадов А. А.,*

*Институт биохимической физики им. Н. М. Эммануэля РАН, Москва,  
e-mail: ak@sky.chph.ras.ru*

*Кукса Ю. И.*

*Институт геоэлектромагнитных исследований РАН, Троицк, Московская обл.,  
e-mail: Kuksa@geo.igemi.troitsk.ru*

### **Абстракт**

Рассмотрены вопросы внедрения методов адаптивной фильтрации на основе технологии MATLAB Web Server в области геофизики, Солнечно-Земной физики и смежных областях знаний для решения широкого круга задач, в том числе прикладных, возникающих при мониторинге состояния околоземного космического пространства, оценке его влияния на технологические системы, биологические объекты и здоровье людей, анализе процессов в земной коре и верхней мантии. Предложена схема организации вычислительного ресурса для специалистов, направленного на обработку больших объемов экспериментальных данных. В его основу положены функционирующие в системе MATLAB прикладные программы адаптивной обработки данных, связь с которыми обеспечивается через Интернет посредством MATLAB Web Server. Для доступа пользователей к вычислительным ресурсам и распределенным в сети Интернет базам данных используются Web-технологии на основе стандартных HTML-форм и документов. Приводятся примеры использования Web-сервиса для исследования геодинамических процессов в Земной коре и верхней мантии на основе вычисления передаточных функций между компонентами магнитотеллурического поля с высоким временным разрешением с целью изучения их временной динамики. Разработаны программы для изучения спектрально-временных и спектрально-пространственных характеристик крупных региональных магнитных аномалий по большому объему данных приземных съемок и спутниковых измерений магнитного поля Земли [1]. Имеются пакеты программ для анализа взаимосвязи между динамикой секторной структуры межпланетного магнитного поля и вариациями магнитного поля



Земли, как в высоких, так и в низких широтах [2]. Общедоступность средства интерактивного анализа данных, организация свободного доступа к ресурсам, исполнение запросов в автоматическом режиме, постоянное пополнение информационной и вычислительной баз гарантируют его развитие и профессиональную востребованность.

Работа выполнена при поддержке РФФИ, грант 03-07-90066.

## **Введение**

Развитие исследований по солнечно-земной физике в значительной степени базируется на знаниях о магнитосфере Земли и ее поведении в периоды магнитных бурь. При этом весь земной шар рассматривается как единая лабораторная установка с системой датчиков в виде отдельных обсерваторий, на основании наблюдений которых мы судим о процессах в околоземном космическом пространстве. В наше время осуществляется широкая международная программа CAWSES «Climate and Weather of the Sun-Earth System» (по-русски «Климат и погода в системе Солнце-Земля»), рассчитанная на 2004-2008 гг. и призванная объединить все исследования по солнечно-земной физике в единый блок на основе опыта предыдущих международных программ. С другой стороны все ведущие космические агентства приняли программу ILWS «International Living With a Star», направленную на координацию спутниковых научных проектов. Целиком программа будет развернута ко времени развития следующего максимума солнечной активности в 2010-2013 гг. В свете подготовки к празднованию 50-летия Международного геофизического года (1957-1958), который фактически положил начало космической эре, учеными разных стран предложено несколько инициативных научных проектов, в том числе «Международный гелиофизический год», см. [www.ihy.gsfc.nasa.gov](http://www.ihy.gsfc.nasa.gov) и «Международный полярный год», см. [www.ipy.gsfc.nasa.gov](http://www.ipy.gsfc.nasa.gov). К этому же времени ожидается запуск сразу 5 микроспутников по проекту THEMIS (Time History of Events and Macroscale Interactions during Substorms, сайт <http://sprg.ssl.berkeley.edu/themis/>), целью которого является исследование суббуревых нестабильностей, формирующихся при прохождении частиц солнечного ветра через магнитопаузу и приводящих к возникновению суббурь. Во всех этих проектах учтены требования наземной поддержки исследований, в первую очередь в виде мониторинга состояния магнитного поля Земли и ионосферы. В международном масштабе основой системы мониторинга состояния магнитного поля Земли служит система «Интермагнит» [www.intermagnet.ru](http://www.intermagnet.ru) объединяющая данные более 90 магнитных обсерваторий со всего мира. В недавнем прошлом собрана база данных геомагнитных обсерваторий РОССИИ за 1984–2000 гг. на CD-ROM и издана с пояснительной брошюрой [3]. CD-ROM размещен на сайте <http://magbase.rssi.ru/index.htm>. Пополнение базы данных основано на

взаимодействии всех институтов, ведущих наблюдения на геомагнитных обсерваториях России. Создается программное обеспечение для работы с базами данных. В рамках гранта РФФИ № 03-07-90066 в ИЗМИРАН ведется отработка технологии интерактивного доступа пользователей к вычислительным ресурсам и распределенным в сети Интернет базам экспериментальных данных на основе средства MATLAB Web Server, использующего возможности Интернет для передачи данных в систему MATLAB для вычислений и отображения результатов в Web-браузерах пользователей с использованием стандартных HTML-форм.

### ***Некоторые аспекты анализа данных в геофизике***

Нередко изучаемые в геофизике процессы, происходящие в природных или природно-технических системах, недоступны для непосредственного наблюдения. В таких ситуациях исследователи вынуждены прибегать к наблюдению других, доступных процессов, порождаемых изучаемыми. Наблюдаемые и изучаемые процессы могут быть связаны достаточно сложным образом, на них могут оказывать влияние иные процессы, происходящие в других системах, они могут быть искажены шумами и т. п. Стремление полнее понять изучаемые явления и повысить достоверность заключений об их природе ведет к расширению круга наблюдений, проведению их на распределенной в пространстве сети станций, т. е. переходу к комплексному анализу. Например, мониторинг геодинамических процессов включает сейсмологические, геодезические, геохимические, гидрогеологические, электромагнитные и т. п. наблюдения. Результатом такого мониторинга, с формальной точки зрения, является совокупность временных рядов разнообразных данных.

В практике мониторинга геодинамических процессов широко используются статистические методы анализа данных. В частности, для выяснения степени взаимосвязи наблюдаемых рядов данных используется корреляционный анализ. Однако он не позволяет установить вид этой связи. Регрессионный анализ, в лучшем случае, позволяет предсказывать значения одного из рядов по синхронно принимаемым значениям других. Между тем в природных процессах, как правило, наблюдаются более сложные формы связей, когда значение одного ряда в какой-либо момент времени  $t$  определяется совокупностью значений другого ряда на некотором отрезке времени. Например, если два ряда  $y(t)$  и  $x(t)$  связаны между собой линейной причинно-следственной связью, то эта связь выражается соотношением свертки:

$$y(t) = \int_0^t g(\tau) \cdot x(t - \tau) d\tau + \Delta y(t), \quad (1)$$

где  $g(\tau)$  — некоторая передаточная функция (точнее — ее импульсная пе-

реходная характеристика), а  $\Delta y(t)$  — невязка, характеризующая точность выполнения сверточной связи. Такое же соотношение имеет место и в том случае, когда оба ряда порождаются (являются следствием) одного и того же основного процесса. Если же основной процесс достаточно сложен и может быть охарактеризован несколькими ( $N$ ) независимо протекающими субпроцессами, то и среди наблюдаемых рядов нередко можно выделить  $N$  независимых, и любой другой ряд, порождаемый тем же основным процессом, будет описываться векторным соотношением свертки:

$$y(t) = \sum_{i=1}^N g_i(t) * x_i(t) \quad (2)$$

На практике соотношения (1) или (2) выполняются с погрешностью  $\Delta y(t)$ , которая включает в себя шумы измерений, неучтенные субпроцессы, а так же возможные влияния на процесс измерений других процессов, сторонних по отношению к изучаемому (например, экзогенных). Их влияние можно исключить с помощью того же аппарата передаточных функций, примененного к иной группе рядов наблюдаемых данных, связанных с этим сторонним процессом. Связи вида (1) и (2) не универсальны. Они имеют место, когда наблюдаемые данные линейно связаны с основным процессом и связи между их рядами стационарны (хотя сами ряды могут и не обладать свойством стационарности). Передаточные функции зависят, как правило, от физико-геологического строения среды в окрестности пункта наблюдения. Если это строение изменяется в силу протекающих в среде процессов, то изменяются и сами передаточные функции. Их изменение несет дополнительную информацию об изучаемых процессах.

### ***Адаптивные методы решения задач геофизики***

Передаточные функции  $g_i(t)$  могут быть найдены путем решения интегральных уравнений свертки (1) или (2) на отрезке времени, на котором заданы значения рядов  $x_i(t)$  и  $y_i(t)$ . После того, как эти функции определены, соотношения (1) и (2) могут использоваться для предсказания поведения ряда  $y(t)$  по известному поведению рядов  $x_i(t)$  (но только, если передаточные функции неизменны во времени). Если значения остаточного ряда  $\Delta y(t)$  превышают измерительные шумы и погрешности вычислений, то он может нести независимую от  $x_i(t)$  информацию. Безотносительно величины  $\Delta y(t)$  полезную независимую информацию могут давать также изменения во времени передаточных функций  $g_i(t)$ .

Решение интегральных уравнений свертки (1) или (2) может быть найдено хорошо известными методами [4], однако для целей мониторинга наиболее подходящими представляются итерационные адаптивные методы [5]. В таких методах последовательно получаемые решения уравнений достигаются при постоянно обновляющихся с течением времени значени-

ях временных рядов  $y(t)$  и  $x_i(t)$ . Если связи между рядами не изменяются, то передаточные функции  $g_i(t)$  принимают после ряда итераций значения близкие к истинным, и в дальнейшем колеблются в их малой окрестности. Если стационарность связей между рядами нарушается, то адаптивные методы позволяют отслеживать изменения передаточных функций во времени. Это, конечно, возможно лишь при более медленных их изменениях по сравнению с вариациями самих рядов.

Наиболее простым и в то же время эффективным методом решения систем алгебраических уравнений, к которым после дискретизации сводятся уравнения свертки, зарекомендовал себя метод наименьших квадратов Уидроу-Хоффа [5]. Итерационный алгоритм этого метода чрезвычайно прост:

$$g_i(k+1, l) = g_i(k, l) + \mu \cdot \Delta y(k) \cdot x_i(k-l), \quad (3)$$

где индекс  $k$  обозначает дискретизированное текущее время наблюдений,  $l$  — время задержки импульсной переходной характеристики, а  $\mu$  — величину параметра сходимости. Значения  $\Delta y(k)$  находятся путем вычитания из наблюдаемого значения  $y(k)$  его синтезированного значения  $y^S(k)$ :

$$y^S(k) = \sum_{i=1}^N \sum_{l=1}^L g_i(k, l) \cdot x_i(k-l) \cdot \Delta t \quad (4)$$

Последнее выражение представляет собой дискретизированную форму соотношения свертки (2) с определенными на  $k$ -ом отрезке времени значениями передаточных функций,  $\Delta t$  — интервал дискретизации.

Характерным примером является обработка данных наблюдений естественных электромагнитных полей Земли с применением описанного выше аппарата передаточных функций. Как известно, эти вариации, в основном, порождаются вариациями токов в ионосфере Земли. В силу удаленности этих токов от точек наблюдения поля на поверхности Земли, их можно рассматривать в первом приближении как токи на плоскости, случайным образом изменяющие свою интенсивность и направление. В соответствии с выбранным представлением систему таких токов можно охарактеризовать двумя независимыми субпроцессами — изменениями составляющих этих токов по двум координатным направлениям на плоскости. Эти токи порождают электромагнитное поле в форме вертикально падающей на Землю плоской волны. Если целью мониторинга является слежение за ионосферными токами, то в наблюдаемом на поверхности Земли поле следует выбрать также две независимо изменяющиеся во времени компоненты электромагнитного поля [6]. В качестве таких компонент естественно выбрать две горизонтальные компоненты геомагнитного поля ( $H_x$ ,  $H_y$ ), на которые электрическое строение Земли оказывает наименьшее воздействие — ввиду большой контрастности по сопротивлению атмосферы и литосферы эти компоненты просто удваиваются по сравнению с падающим полем. Остальные компоненты электромагнитного поля тесно

связаны с геоэлектрическим разрезом (электрическое поле  $E_x$ ,  $E_y$ ) или появляются в силу его горизонтальной неоднородности (вертикальная компонента магнитного поля  $H_z$ ). В рамках сформулированной модели вариации этих компонент в соответствии с выражением (2) могут быть описаны соотношением:

$$H_z(t) = H_x(t) * I_{zx}(t) + H_y(t) * I_{zy}(t) + \Delta H_z(t), \quad (5)$$

где  $I_{zx}(t)$  и  $I_{zy}(t)$  — компоненты индукционного вектора или, в более общей трактовке, — импульсные переходные характеристики соответствующих передаточных функций, отражающие электрическое строение Земли в окрестности точки наблюдения. В вариациях остаточного поля ( $\Delta H_z$ ) в значительной степени ослаблена зависимость от ионосферных токов. Существенный вклад в них вносят поля внутреннего, геодинамического происхождения, возникающие в недрах Земли в результате разнообразных механоэлектрических преобразований. Если в результате каких-либо геодинамических процессов не только генерируются электромагнитные поля, но и изменяется геоэлектрическое строение среды, то соответственно изменяются и передаточные функции  $I_{ij}$ . Их изменения несут независимую по сравнению с остаточными полями информацию о протекающих внутри Земли процессах, которая может быть эффективно использована для целей геодинамического мониторинга.

### **Примеры реализации вычислительного алгоритма**

Блок-схема вычислительного алгоритма метода наименьших квадратов Уидроу-Хоффа представлена на рис. 1.

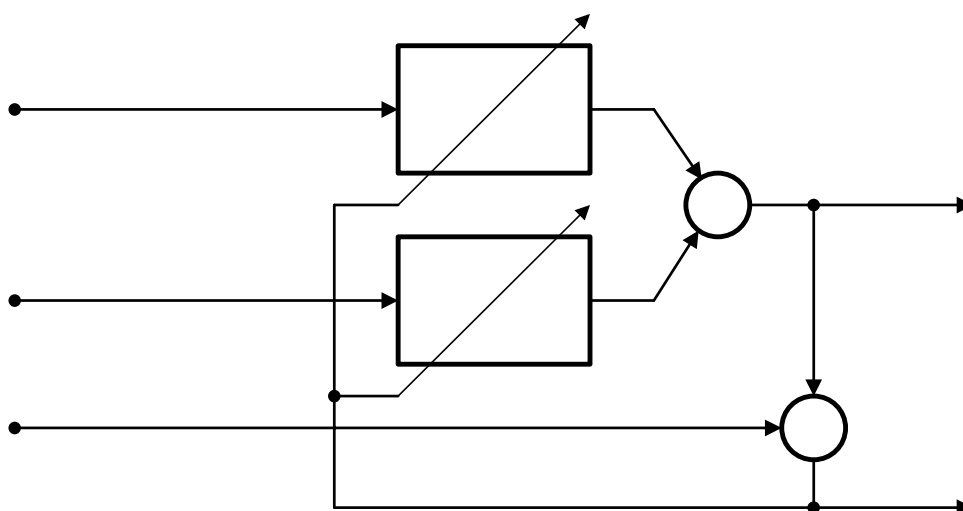


Рис. 1. Блок-схема вычислительного алгоритма.

На этой схеме блоки «Адаптивный фильтр» 1 и 2 реализуют итерационный алгоритм (3). В соответствии с выражением (5) в качестве опорных сигналов на входах блоков используются две горизонтальные компоненты геомагнитного поля ( $H_x$ ,  $H_y$ ), а в качестве основного — вертикальная компонента  $H_z$ . Разностный сигнал на выходе находится путем вычитания из наблюдаемого значения  $H_z$  его синтезированного значения в соответствии с выражением (4).

Рассмотренный алгоритм адаптивного фильтра был применен нами для анализа данных магнитотеллурического мониторинга, проводимого на Бишкекском геофизическом полигоне в начале 90-х годов [7]. С помощью пакета программ, написанных на Паскале, были обработаны ряды данных непрерывной в течение нескольких лет регистрации электромагнитного поля Земли, и графики, характеризующие динамику передаточных функций во времени, были сопоставлены с данными о сейсмической активности региона. На основании полученных результатов были отработаны методики мониторинга сейсмической активности.

В пакете расширения Filter Design, начиная с версии 2.1, входящей в поставку MATLAB 6.1, содержатся функции, реализующие алгоритмы адаптивной фильтрации [8]. Однако, по целому ряду соображений, нами были использованы функции собственной разработки. Код основной MATLAB-программы, реализующей алгоритм метода наименьших квадратов Уидроу-Хоффа, приведен ниже:

```
k=1:L;
Ys = sum(sum(W(:,k).*X(:,i-k+1)));
E(i) = S(i)-Ys;
for j=1:Nref
    W(j,k)=W(j,k)+2*U(j)*E(i)*X(j,i-k+1);
end;
```

Здесь:  $L$  — длина фильтра;  $N_{\text{ref}}$  — число опорных сигналов;  $E$  — разностный сигнал;  $Y_s$  — синтезированный сигнал;  $W$  — передаточные функции. Разработанная нами программа позволяет использовать любое число опорных сигналов и произвольное количество циклов адаптации.

Новые возможности, предоставляемые пакетом MATLAB, позволили провести дополнительную обработку данных. На Рис. 1. показан пример вычисления передаточных функций между компонентами магнитного поля Земли в соответствии с выражением (5) продолжительностью в 24 часа. В качестве основного сигнала принята вертикальная компонента вариаций геомагнитного поля, в качестве опорных — горизонтальные компоненты. На графиках представлены исходный основной временной ряд (Raw data), разностный временной ряд (Residuals), а также передаточные функции между вертикальной и горизонтальными компонентами.

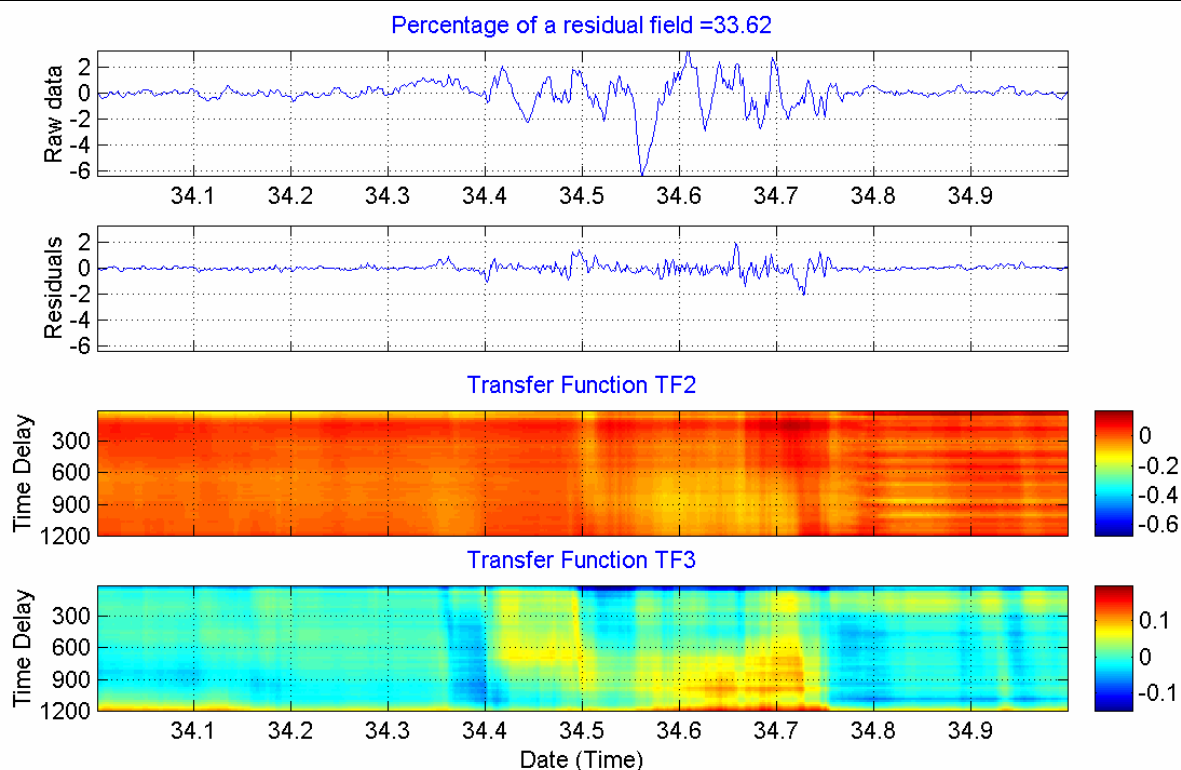


Рис. 1. Пример вычисления передаточных функций между компонентами электромагнитного поля Земли: 1 — основная компонента  $H_z$ ; 2 — остаточное поле; 3–4 — передаточные функции по компонентам  $H_x$ ,  $H_y$ .

Из графика видно, как в течение суток меняется характер вариаций электромагнитного поля, а также характер связи между компонентами, что отражается в изменениях передаточных функций. Детальный анализ физического смысла этих изменений выходит за рамки данной работы.

### ***Спектрально-временной анализ узкополосным адаптивным фильтром***

Определенный интерес представляет анализ изменений спектра колебаний в разностном (остаточном) поле. Учитывая специфику геофизических исследований, а именно, интерес к низкочастотной части спектра в диапазоне периодов от единиц минут до часа, пришлось отказаться от традиционных методов спектрального анализа. Для спектрально-временного анализа рядов данных нами был использован метод выделения гармонических составляющих с помощью узкополосных адаптивных режекторных фильтров [9]. Их преимущества заключаются в простоте перестройки полосы пропускания, практически неограниченном подавлении соседних гармоник и точном слежении за частотой. Принцип их действия может быть описан той же блок-схемой рис.1. Отличие заключается лишь в том, что на входы адаптивных фильтров 1 и 2 в качестве опорных сигналов подаются две квадратурные составляющие моногармонического сигнала с

частотой  $\omega_0$ , равной частоте выделяемой гармоники. Анализируемый временной ряд  $S_i$  поступает на вход основного сигнала. Формируемые в процессе вычислений квадратурные составляющие  $x_{1i}$  и  $x_{2i}$  в каждой точке анализируемого временного ряда принимают значения:

$$\begin{aligned}x_{1i} &= C \cdot \cos(\omega_0 i \Delta t + \varphi) \\x_{2i} &= C \cdot \sin(\omega_0 i \Delta t + \varphi)\end{aligned}\quad (6)$$

где  $C$  — амплитуда колебания,  $\Delta t$  — шаг дискретизации. Из этих значений и весовых коэффициентов адаптивного фильтра  $W_{1i}$  и  $W_{2i}$  формируется синтезированный сигнал  $S_{si} = x_{1i}W_{1i} + x_{2i}W_{2i}$ . Разность между исходным и синтезированным сигналами  $\varepsilon_i = S_i - S_{si}$  используется в качестве параметров, управляющего обновлением весовых коэффициентов фильтра на каждом шаге вычисления в соответствии с выражениями:

$$\begin{aligned}W_{1,i+1} &= W_{1,i} + 2\mu\varepsilon_i x_{1i} \\W_{2,i+1} &= W_{2,i} + 2\mu\varepsilon_i x_{2i}\end{aligned}\quad (7)$$

где  $\mu$  — параметр, характеризующий скорость сходимости алгоритма адаптации коэффициентов фильтра. Добротность такого фильтра определяется выражением  $Q = \omega_0 \Delta t / (2\mu C^2)$ . Синтезированный сигнал  $S_{si}$  является выделяемой гармоникой с частотой  $\omega_0$ , огибающая которой находится по формуле:

$$A_i = C \sqrt{W_{1i}^2 + W_{2i}^2} \quad (8)$$

Код MATLAB-программы узкополосного адаптивного режекторного фильтра в соответствии с выражениями (6) и (7), приведен ниже:

```
Xs1 = Amax.*sin(pii.*step);
Xs2 = Amax.*cos(pii.*step);
step = step+1;
Ys = Ws1.*Xs1+Ws2.*Xs2;
Eps = AvErr(i)-Ys;
Ws1 = Ws1 + (2*Eps).*Xs1;
Ws2 = Ws2 + (2*Eps).*Xs2;
```

При помощи такого фильтра были обработаны данные временного ряда остаточного поля из предыдущего примера. Результат их обработки в виде динамического спектра показан на рис. 2. Динамические спектры вариаций позволяют детально исследовать наличие и амплитуды гармонических составляющих временного ряда в диапазоне периодов от 2 мин. до 1 часа с высоким разрешением. При этом не имеет значения, какой шаг будет установлен между соседними гармониками, а также характер шкалы — линейный, логарифмический, либо произвольный переменный.



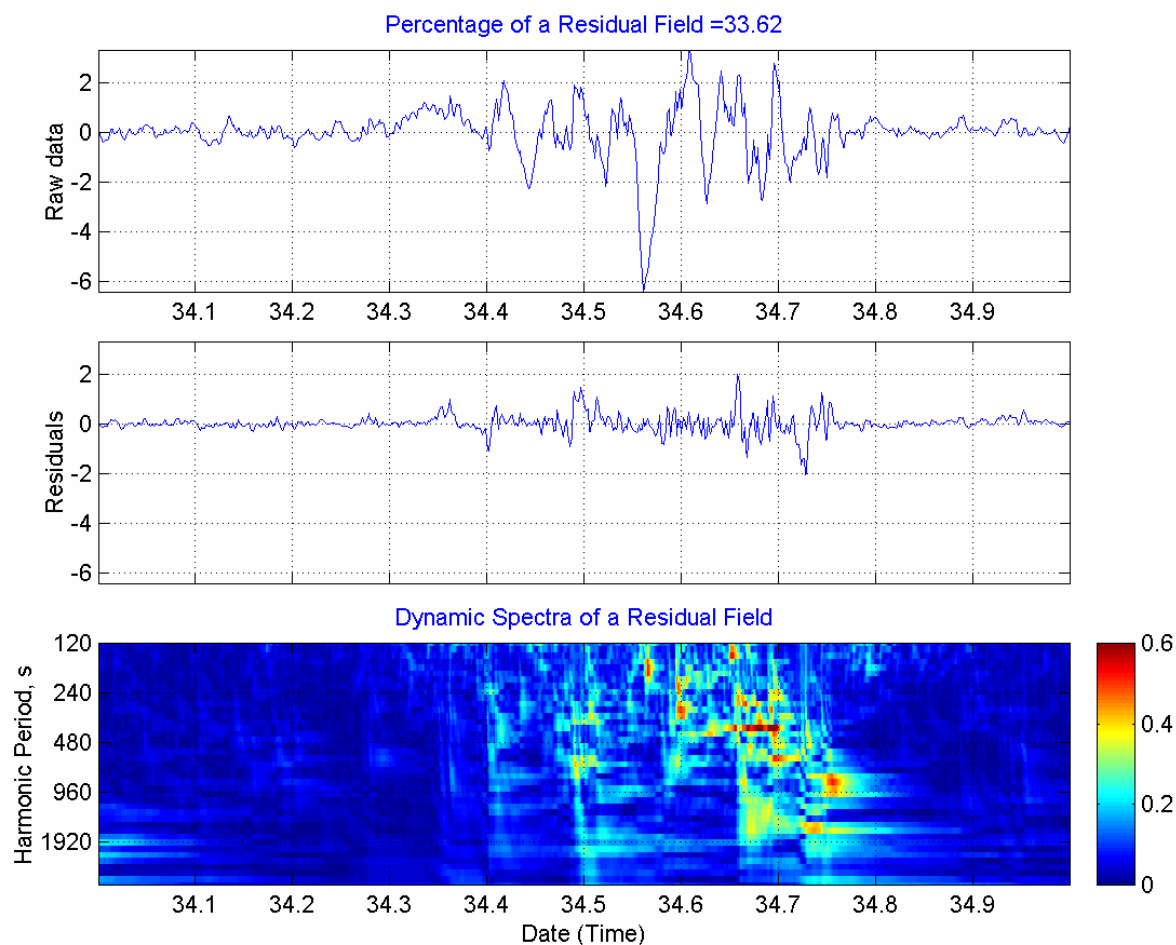


Рис. 2. Динамический спектр вариаций остаточного поля после вычисления передаточных функций между компонентами электромагнитного поля Земли.

### ***Инструментальное средство MATLAB Web Server***

Особый интерес в геофизике представляет сетевой доступ к данным, распределенным в Интернет, и интерактивный анализ экспериментальных данных по запросам пользователей. Процедура должна быть максимально упрощена, с тем, чтобы обеспечить эффективное обращение к данным и вычислительными ресурсам даже для той категории пользователей, которые не имеют достаточного опыта создания и эксплуатации программ обработки данных, получаемых в процессе геофизических экспериментов.

Одним из вариантов обеспечения интерактивного взаимодействия может быть применение Web Server из пакета MATLAB для передачи данных на главный компьютер с установленной системой MATLAB, проведения на нем вычислений и отображения результатов на Web-браузере пользователя путем стандартной Web-технологии в виде HTML-форм и документов. Форма дает пользователю возможность проверить наличие данных, выбрать требуемую обсерваторию, интервал времени, компоненту магнитного поля, воздействовать на параметры рисунка (цветной либо мо-

нохромный) и т. п. На Рис. 3. показана HTML-форма для отправки запроса на обработку заданного отрезка временного ряда.

The screenshot shows a Microsoft Internet Explorer window titled "Dynamic Spectra in an HTML Page". The address bar shows the URL <http://fenix.izmiran.rssi.ru/svantoweb.html>. The page content includes a navigation bar with buttons: "MAG-files Data Plot", "Spectral-Time Analysis", "Data Availability Table", and "Home". Below this is a logo for "MATH BY MATLAB" and the text "Mag Data Dynamic Spectra display in an HTML Page". The form contains several sections:

- DATE & OBSERVATORY CODE** (Check the data availability): Year: 2003, Month: Nov, Code: MOS.
- START TIME [UT]**: Day: 20 + 0, Hour: 00 + 3.
- DURATION (Maximum: 3 days)**: Day(s): 00 + 1, Hour(s): 00 + 9.
- COMPONENT**: Radio buttons for H, D, and Z.
- PRINTER**: Radio buttons for Color and B/W.
- Buttons: Submit, Reset.
- Footer: ©2002 by The MathWorks, Inc. All rights reserved. MATLAB is a registered trademark of The MathWorks, Inc.

Рис. 3. Пример HTML-формы для отправки запроса на обработку данных.

В ответ на запрос пользователь получает результат обработки конкретного временного ряда в виде графического представления или сформированного нового файла, содержащего как результаты обработки, так и подготовленный новый ряд данных для последующей обработки средствами пользователя.

На этом примере показана возможность организации интерактивного доступа к базе данных геомагнитных обсерваторий РОССИИ за 1984 — 2000 гг., размещенной на сайте <http://magbase.rssi.ru/index.htm>, а также данным непрерывной регистрации вариаций геомагнитного поля в обсерватории Москва [http://www.izmiran.rssi.ru/magnetism/mos\\_data.htm](http://www.izmiran.rssi.ru/magnetism/mos_data.htm). На Рис. 4 приведен график вариаций одной из компонент геомагнитного поля во время большой магнитной бури 20 ноября 2003 г., сформированный по за-

просу пользователя, а также результат спектрально-временного анализа этих вариаций (динамические спектры).

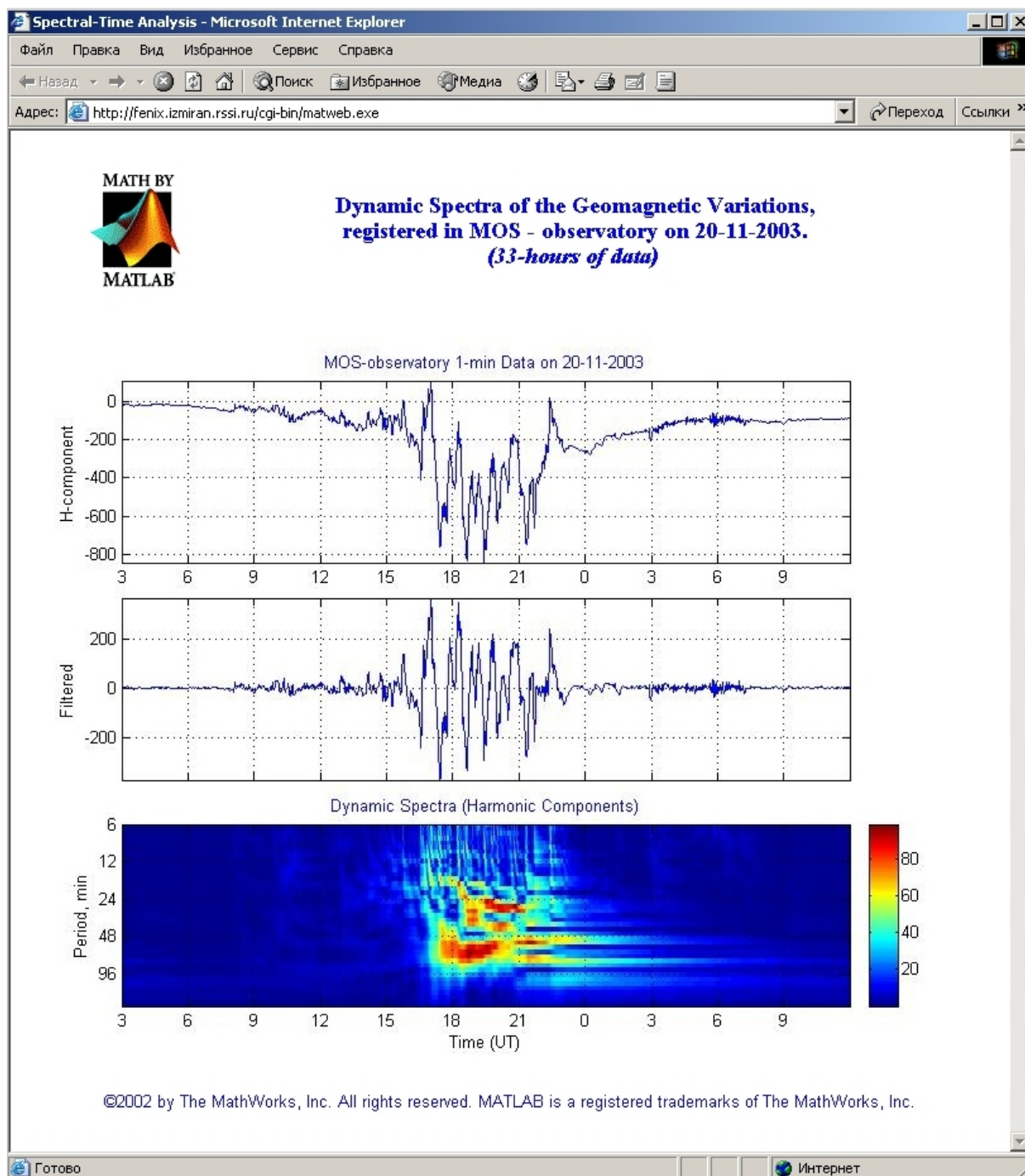


Рис. 4. Пример формы, возвращаемой в ответ на запрос пользователя.

Для пересылки данных с удаленных сайтов Интернет используются либо средства MATLAB, либо небольшие CGI-программы на языке Perl. Фрагмент кода MATLAB-программы для запроса данных с удаленного сайта приводится ниже:



```

fn = strcat('V',YeMo,',' ,Cod);
pn = strcat('D:\CDMASTER\DATABASE\',Yea,'\ ',YeMo,'\ ');
if exist([pn,fn]) ~= 0
    fin = fopen([pn,fn], 'r');
else
    if exist(fn) == 0
        pn = strcat('http://magbase.rssi.ru/database/',Yea,'\ ',YeMo,'\ ');
        URLWRITE([pn,fn],fn); end
    fin = fopen(fn, 'r');
end;

```

В этой программе из данных формы запроса, полученной Web Server, формируется имя запрошенного файла и путь к нему (в соответствии с принятой структурой базы данных). Затем следует проверка на наличие данных непосредственно на той машине, где установлена MATLAB и исполняемые программы обработки. В случае отсутствия данных производится их поиск по всем известным адресам в Интернет.

### **Заключение**

Само по себе представление геофизических данных хотя и является крайне важным шагом на пути их активного использования, но не может удовлетворить всем современным требованиям ученых-исследователей. Перспективным направлением в этом плане может быть создание обобщенных аналитических представлений — например, вариационных магнитных данных в виде моделей эквивалентных токовых систем и других параметров ионосферы в планетарном масштабе. В последние годы эти методы анализа и представления данных были развиты и реализованы в виде сервиса в реальном времени AMIE — «Assimilative Mapping of Ionospheric Electrodynamics», на сайте <http://amie.ngdc.noaa.gov/current.html>. Такое обобщенное представление большого объема данных позволяет сразу перейти от качественного описания развития возмущений магнитного поля к их количественной оценке. Применение новейших информационных технологий открывает широкому кругу специалистов доступ через Интернет к базам данных и их интерактивному анализу средствами системы MATLAB. Пример использования MATLAB Web Server для доступа к «Базе данных геомагнитных обсерваторий России» на CD-ROM и вычисления спектрально-временных характеристик геомагнитных вариаций по данным этой базы представлен в тестовой версии на сайте <http://vlod.izmiran.rssi.ru>.

Работа выполнена при поддержке РФФИ, грант 03-07-90066.

### **Литература**

1. Ротанова Н. М., Цветков Ю. П., Одинцов В. И., Бурцева Э. А. Магнитное сканирование земной коры Сибирского региона с борта страто-

- сферного аэростата // Исследование Земли из космоса.— 2001.— №4.— С.63–73.
2. *Зайцев А. Н., Одинцов В. И.* Наблюдение эффектов секторной структуры ММП в околоземном пространстве по геомагнитным данным // Актуальные проблемы физики Солнечной и звездной активности. Конф. стран СНГ и Прибалтики (Нижний Новгород, 2–7 июня 2003): Сб. докл. в 2-х т. Т. II.— Нижний Новгород: ИПФ РАН, 2003.— С.460–464.
  3. *Амиантов А. С., Зайцев А. Н., Одинцов В. И., Петров В. Г.* Вариации магнитного поля Земли: База цифровых данных магнитных обсерваторий России за период 1984–2000. (брошюра и оптический диск CD-ROM).— М.: СтройАрт, 2001.— 52 стр.
  4. *Тихонов Н. Н., Арсенин В. Я.* Методы решения некорректных задач.— М.: Наука, 1979.— 285 с.
  5. *Уидроу Б., Стирнз С.* Адаптивная обработка сигналов.— М., Радио и связь, 1989.— 440 с.
  6. *Светов Б. С., Шимелевич М. И.* Определение линейных связей между компонентами магнитотеллурического поля — основные принципы // Физика Земли.— 1982.— №5.— С.59–67.
  7. *Светов Б. С., Каринский С. Д., Кукса Ю. И., Одинцов В. И.* Магнитотеллурический мониторинг геодинамических процессов // Физика Земли.— 1997.— №5.— С.36–46.
  8. *Сергиенко А. Б.* Алгоритмы адаптивной фильтрации: особенности реализации в MATLAB // Exponenta Pro. Математика в приложениях.— 2003.— №1.— С.18–28.
  9. *Уидроу Б., Гловер Д. Р., Макул Д. М. и др.* Адаптивные компенсаторы помех. Принципы построения и применения // ТИИЭР.— 1975.— Т.63.— №12.— С.69–98.

УДК 004

## **ПРИМЕНЕНИЕ ИНТЕРАКТИВНОГО ИНСТРУМЕНТАРИЯ FILTER DESIGN&ANALYSIS TOOL В ЛАБОРАТОРНОМ ПРАКТИКУМЕ**

*Резванов Д. Х.*

*Московский государственный технологический университет «СТАНКИН», Москва,  
e-mail:rdx@forte-it.ru*

В эпоху всеобщей компьютеризации, автоматизации и необходимости сокращения сроков выполнения работ, неотъемлемой частью разработки современных приборов и информационно-измерительных систем становится система автоматизированного проектирования. Важным аспектом подготовки высококвалифицированных инженеров является повышение их компьютерной грамотности, в том числе обучение навыкам работы в среде различных пакетов программ автоматизированного проектирования, и в частности алгоритмам и методикам решения прикладных инженерных задач с использованием различных систем САПР. Кроме того, компьютерные технологии и системы САПР давно оправдали свою эффективность при использовании в учебном процессе, благодаря их использованию значительно сократилось время выполнения рутинных операций, характерных для аналитических расчетов, таким образом больше времени уделяется изучению предметной области.

Одной из наиболее апробированных и широко используемых систем автоматизированного проектирования в процессе обучения является система MATLAB, зарекомендовавшая себя как эффективный инструмент для проведения лабораторных и практических занятий.

Ряд лабораторных работ при подготовке специалистов по направлению «Приборостроение» целесообразно основать на применении интерактивного инструментария FDATool (Filter Design & Analysis Tool), который является GUI (Graphic User Interface) приложением пакета программ MATLAB, предоставляет широкие возможности для анализа, в том числе эффектов квантования, синтеза и построения цифровых фильтров, как с фиксированной так и с плавающей точкой, реализующих практически любой современный алгоритм работы, обладает дружелюбным интерфейсом и богатыми возможностями визуализации результатов работы. Приложение FDATool сочетает в себе функции пакетов расширения Signal Processing Toolbox и Filter Design Toolbox, а также является одним из блоков пакета моделирования Simulink, что позволяет экспортировать спроектированный цифровой фильтр в Simulink модели. Интеграция FDATool с соответствующими компонентами системы MATLAB через Code Composer

Studio позволяет генерировать С код для цифровых процессоров, либо осуществлять запись программы реализующей алгоритм работы спроектированного цифрового фильтра непосредственно в память процессоров таких производителей как Texas Instruments, Motorola и др., при наличии соответствующих пакетов расширения.

Логика работы с приложением FDATool подразумевает интерактивный режим, то есть использование функций командной строки путем активации и заполнения соответствующих полей инструментария. Такой метод работы с системой зарекомендовал себя как наиболее подходящий для использования в учебном процессе и вызывает живой интерес у студентов.

В настоящее время планируется использование приложения FDATool при проведении лабораторных работ на факультете «Метрологической информатики» МГТУ «СТАНКИН».

Приложение FDATool может быть рекомендовано при проведении лабораторных работ по цифровой обработке измерительной информации и позволяет: получать частотные и временные характеристики цифровых фильтров; изучать структуры различных цифровых фильтров; производить анализ эффектов квантования.

Дальнейшие работы планируется направить на совершенствование и углубление методического обеспечения лабораторного практикума по циклу учебных дисциплин связанных с цифровой обработкой сигналов.

Лабораторным работам предшествует курс лекций и семинарские занятия, где студенты получают необходимые сведения о системе MATLAB и методические рекомендации по работе с приложением FDATool.

### ***Литература***

1. Сергиенко А. Б. Цифровая обработка сигналов.— СПб.: Питер, 2002.— 603 с.
2. Дорф Р., Бишоп Р. Современные системы управления.— М.: Москва лаборатория базовых знаний «ЮНИМЕДИАСТАЙЛ», 2002.— 831 с.
3. Дьяконов В., Абраменкова И. MATLAB обработка сигналов и изображений.— СПб.: Питер, 2002.— 602 с.

УДК 517.9

## ОПЫТ ИСПОЛЬЗОВАНИЯ MATLAB ДЛЯ РАСЧЕТА ПАРАМЕТРОВ ПИД-РЕГУЛЯТОРА

Стрелковский Н. Н.

Московская государственная академия приборостроения и информатики

e-mail: main\_mail@front.ru

Цель работы — разработка ПИД-регулятора для технологического процесса приготовления корма для животных (рис. 1).

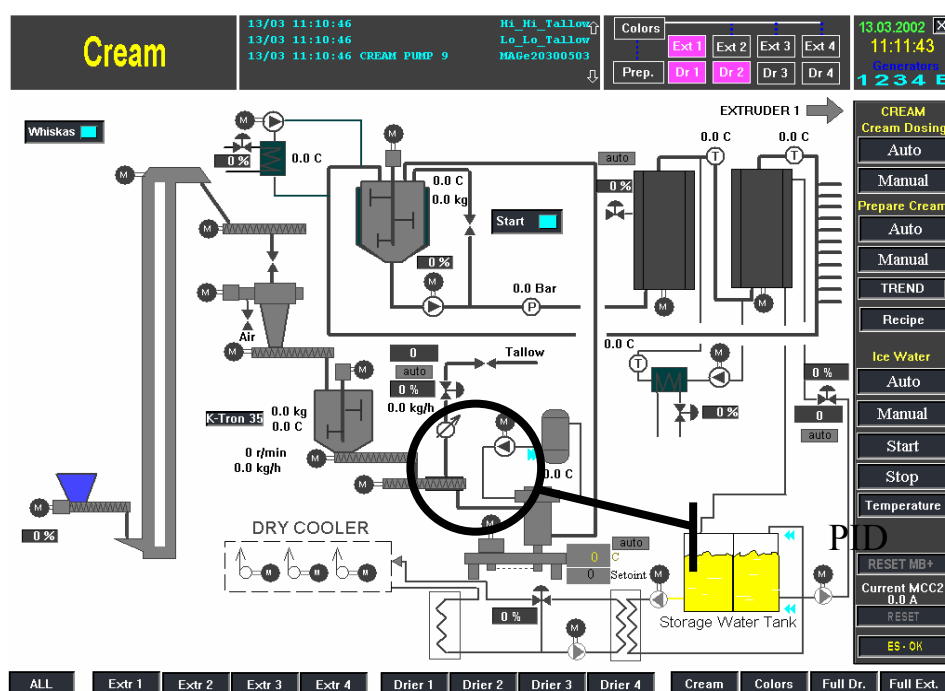


Рис. 1. Технологический процесс.

Для разработки регулятора необходимо построить математическую модель, адекватную реальной системе (на рис. 1 обведено кругом). Входной параметр  $u(t)$  — состояние клапана (%); выходной —  $y(t)$  — поток жира (кг/ч). Для построения математической модели использовался System Identification Toolbox. Моделб ARXQS наиболее адекватна — 88% (рис. 2). Передаточная функция идентифицируемой системы в переходном режиме:

$$\frac{0.2826s^4 - 0.006512s^3 + 0.009636s^2 - 0.005273s - 4.585e^{-20}}{s^4 - 1.116s^3 + 0.313s^2 - 0.125s - 0.3679}$$

Simulink-модель объекта управления представлена на рис. 3.



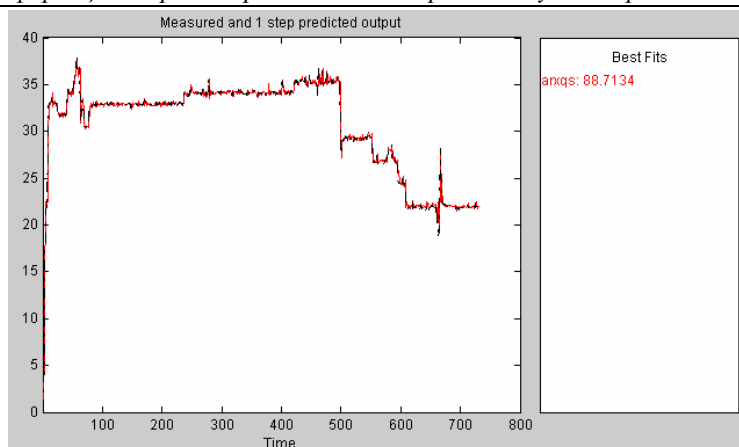


Рис. 2. Выходы объекта и модели.

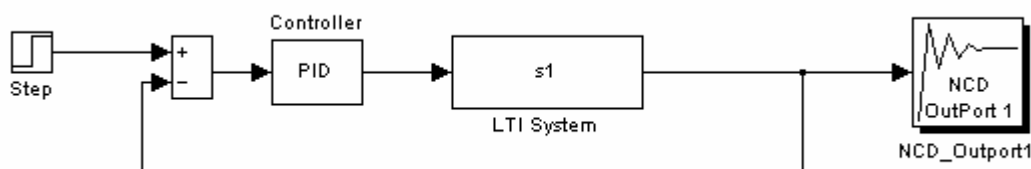


Рис. 3. Simulink-модель.

При оптимизации в NCD получены оптимальные значения ПИД-регулятора  $K_p = 102.13$ ;  $K_i = 87.104$ ;  $K_d = 33.715$  (рис. 4).

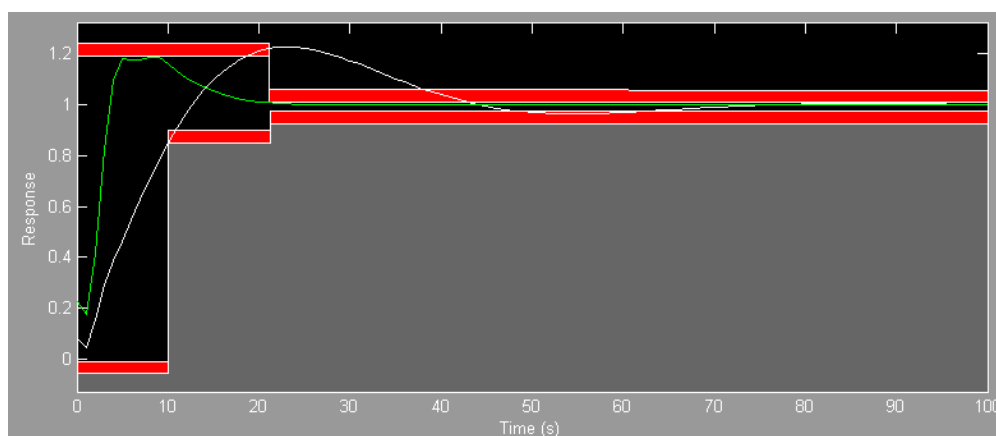


Рис. 4. Переходной процесс.

Полученные результаты использовались для реальной системы при управлении технологическим процессом котроллером Quantum Modicon. В программном обеспечение ModSoft аппаратно реализована функция PID–2.

УДК 519.6

## КОМПЬЮТЕРНЫЙ ПРАКТИКУМ ПО КУРСУ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ

*Тарасевич Ю. Ю., Пономарева И. С.*

*Астраханский государственный университет, Астрахань,  
e-mail: tarasevich@astranet.ru*

У студентов 5 курса спросили: «Стоит ли значительную часть семестра изучать интегрирование методом неопределенных коэффициентов, с помощью универсальной тригонометрической подстановки и по частям, или лучше за это же самое время освоить технику вычисления интегралов с помощью какого-либо пакета компьютерной алгебры, например, Maple или Mathematica?» Вопрос был задан на факультете дополнительного образования, и поэтому группа была сборной: половина — студенты-математики, половина — студенты-информатики. Мнения разделились поровну: «Несомненно, учить технике вычисления интегралов», — сказали математики. «Конечно, осваивать пакет компьютерной алгебры», — без тени сомнения ответили информатики. Как это обычно бывает, истина, вероятно, находится где-то между этими крайними точками зрения. Специалисты, обучавшиеся «чистой» математике обычно настаивают на том, что математика формирует логическое мышление. С этим трудно спорить, но можно добавить, что логическое мышление может сформироваться благодаря программированию, игре в шахматы или в столь любимый математиками преферанс. Прекрасно владевший математикой академик Л.Д. Ландау писал о преподавании математики: «Мне не хочется дискутировать с достойной средневековой схоластики мыслью, что путем изучения ненужных им вещей люди будто бы научаются логически мыслить» [1]. Изучив курс высшей математики, студенты порой оказываются совершенно беспомощными при решении конкретных практических задач. Например, на олимпиаде по компьютерной физике, студент-физик 4 курса написал программку на Паскале, реализующую метод Эйлера для решения дифференциального уравнения автоколебательного процесса. Лет 20 назад это, наверное, можно было бы приветствовать, но сегодня, когда широко распространены самые разнообразные математические пакеты, способные с высокой точностью решать дифференциальные уравнения, описанную ситуацию иначе как курьезной не назовешь, поскольку участникам Олимпиады разрешалось использовать любое доступное программное обеспечение. Представьте, что житель современного города, когда захочет напиться, возьмет лопату, выйдет на улицу и начнет копать колодец. Ситуация абсурдна, но именно так поступил студент, написав при-

митивную реализацию неэффективного алгоритма для решения типовой задачи.

Система образования — система с колоссальным последствием: последствия образовательных реформ и экспериментов будут проследиваться еще в 2–3 поколениях. Именно поэтому система образования обязана быть чрезвычайно консервативной. Если ошибки врачей видны почти сразу и их последствия могут быть трагичными, то дефекты образования сказываются не сразу, но последствия их могут быть катастрофичными для общества. В то же самое время, система образования не может не эволюционировать: на смену счетам и таблицам Брадиса пришли логарифмические линейки, затем калькуляторы и, наконец, компьютеры. Это — естественный процесс обновления, отражающий изменения реальной жизни. Однако многие подходы в системе высшего образования остаются очень архаичными и явно не соответствуют возможностям и потребностям времени.

Образование должно преследовать как прагматические, так и мировоззренческие цели. Если человек после многих лет изучения физики в школе и вузе не в состоянии измерить напряжение в розетке и охотно проглатывает всевозможную антинаучную околесицу, льющуюся на него из средств массовой информации, это означает, что ни прагматические, ни мировоззренческие цели не достигнуты.

Широкое применение при изучении математических и естественнонаучных дисциплин математических пакетов, в частности, пакета MATLAB, помогает перенести акцент в обучении на принципиальные вопросы, отдав «технические» задачи компьютеру.

Ранее мы уже писали об опыте применения MATLAB при изучении численных методов и математического моделирования [2]. Нами разработан комплекс программ на MATLAB, с помощью которого можно исследовать целый ряд моделей, ставших классическими. Перечень лабораторных работ соответствует учебному пособию [3]. Новая версия комплекса реализована на MATLAB 6.1. По сравнению с предыдущей версией комплекс существенно переработан: теперь каждая программа состоит только из 3 файлов: файла графического интерфейса, собственно программы и, если это предусмотрено, файла Simulink для проведения имитационного моделирования. Каждая лабораторная работа имеет краткую теоретическую справку о модели. Файлы справки подготовлены в формате pdf, чтобы сделать комплекс максимально независимым от используемой операционной системы. Основное назначение комплекса — дать студентам возможность сосредоточиться на содержательной части модели, изучить качественные особенности ее поведения, не отвлекаясь на техническую часть, связанную с реализацией модели на компьютере. Такой подход позволяет сформировать у студентов то, что сейчас принято называть «нелинейным мышлением» [4].

Все модели имеют удобный графический интерфейс с различными элементами управления, с помощью которых пользователь имеет возможность изменять параметры моделей, что позволяет анализировать ее поведение при различных условиях.

Первая часть комплекса — «Дифференциальные модели». Пользователь имеет возможность изучить ряд базовых моделей физики, химии, биологии:

1. Нелинейный маятник;
2. Модифицированная модель Вольтерры–Лотки (модель хищник–жертва);
3. Модель межвидовой конкуренции;
4. Модель Холлинга–Тэннера;
5. Модель Лоренца;
6. Модель Ресслера;
7. Генератор Ван-дер-Поля;
8. Тримолекулярная модель (брюсселятор).

При построении модели пользователь может не только задавать различные коэффициенты дифференциального уравнения, описывающие данную модель, но и начальные условия, непосредственно выбирая их на координатной плоскости. При этом фазовый портрет изображается вместе с особыми линиями, а также указывается тип особых точек (рис.1).

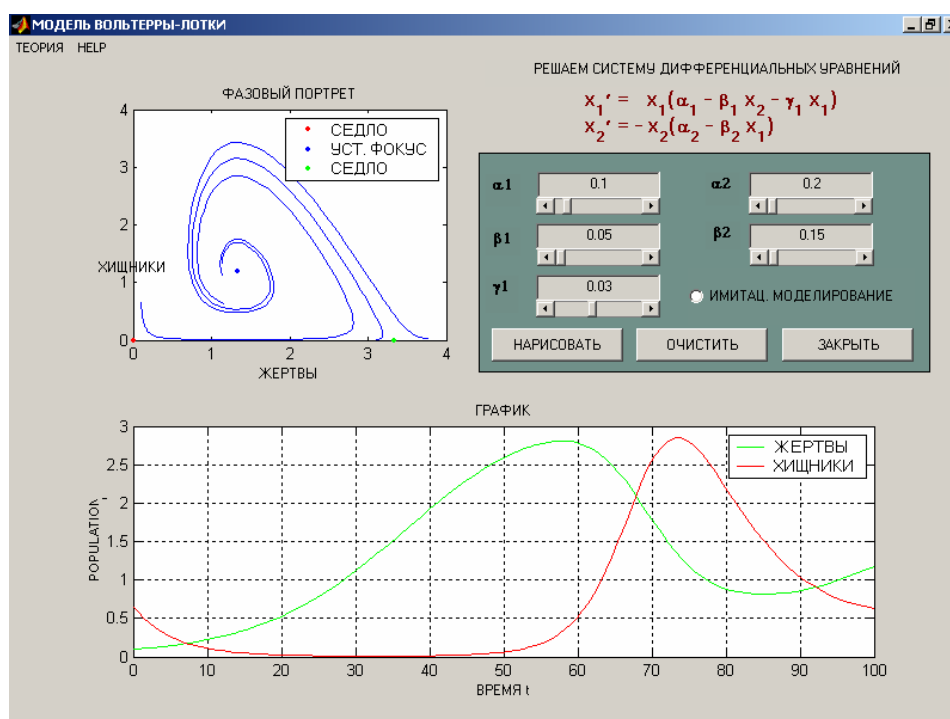


Рис.1. Интерфейс модифицированной модели Вольтерры–Лотки.

«Имитационное моделирование» — обязательный элемент на всех электронных страницах данного раздела лаборатории, который открывает дополнительное окно Simulink, отображающее схему данной модели.

Вторая часть — «Отображения» (рис. 2). Рассматриваются как одномерные, так и двумерные отображения:

1. Логистическое отображение;
2. Динамика популяции;
3. Динамика роста популяции, ограниченной эпидемиями;
4. Диссипативное и консервативное отображения Фейгенбаума;
5. Диссипативное и консервативное отображения Хенона.

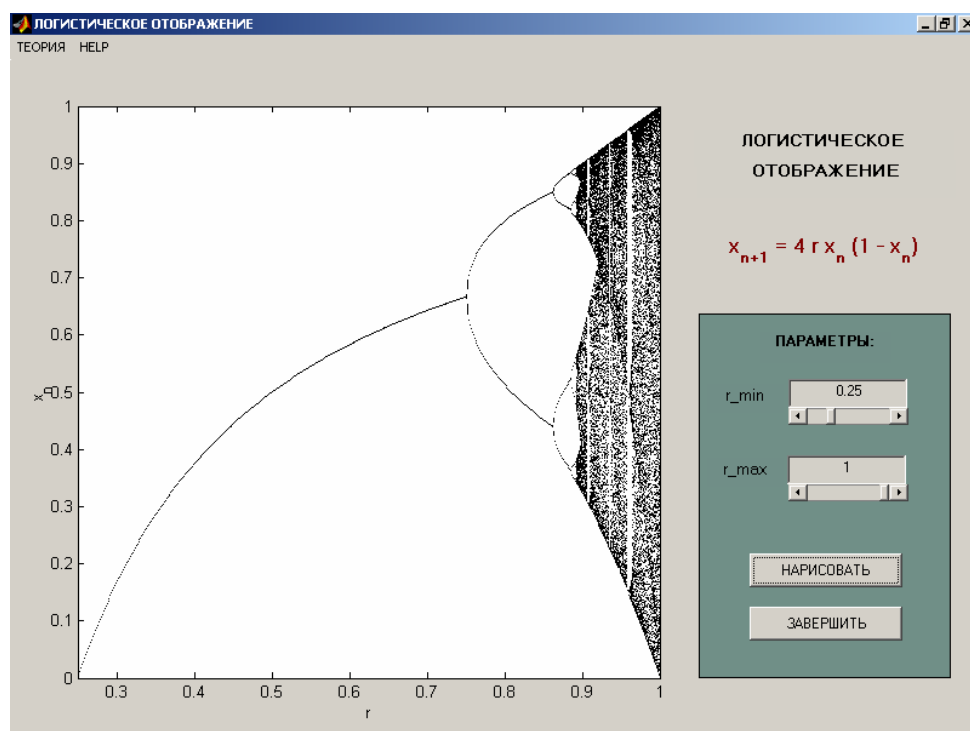


Рис. 2. Интерфейс модели «Логистическое отображение».

Третья часть — «Клеточные автоматы» (рис.3). Здесь пользователь имеет возможность исследовать поведение классических моделей с дискретным пространством и временем в зависимости от параметров и начальной конфигурации. В настоящее время реализованы следующие клеточные автоматы:

1. Игра «Жизнь».
2. Модель Винера–Розенблюта.

Во втором семестре студенты получают задание самостоятельно реализовать ту или иную модель. Здесь задача уже совершенно иная — студентам требуется соединить знания и навыки, полученные при изучении математики, программирования, пакетов прикладных программ. Студентам для разработки программы разрешается использовать любое известное им

инструментальное средство. Поскольку практические занятия по курсу численных методов уже несколько лет проводятся с ориентацией на MATLAB, многие студенты для реализации моделей используют именно этот пакет.

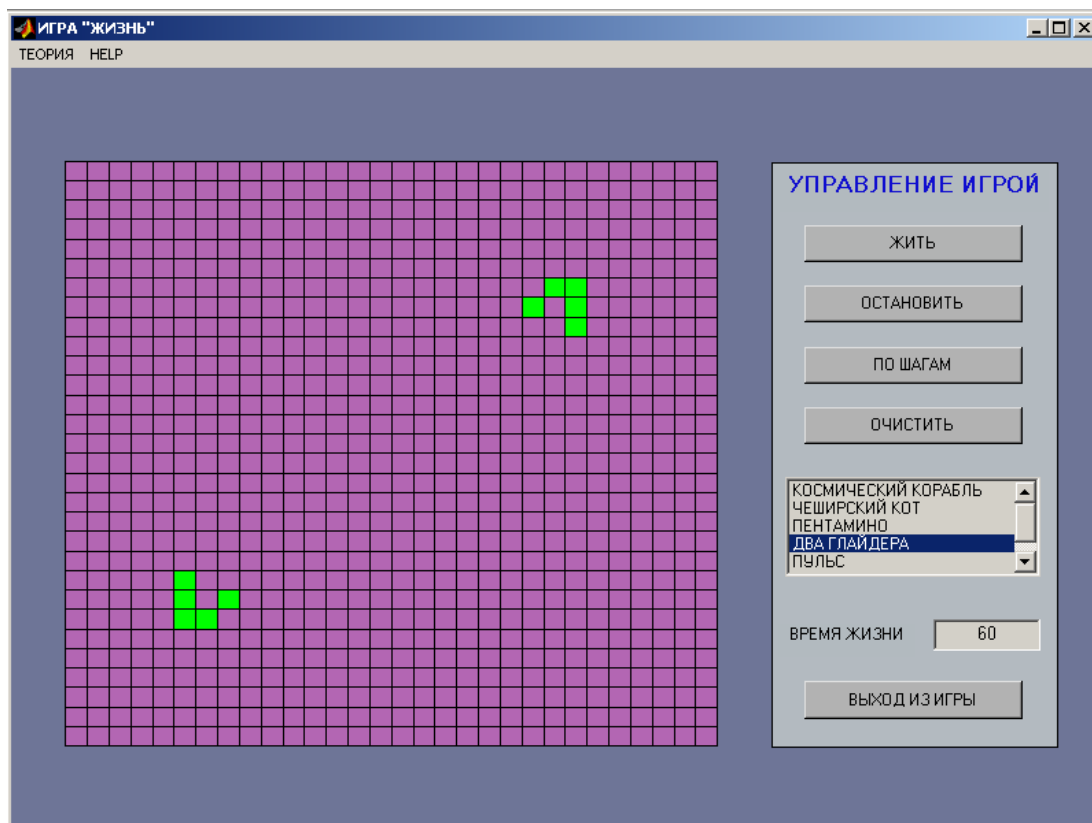


Рис. 3. Интерфейс модели «Логистическое отображение».

### Литература

1. Лифшиц Е.М. Живая речь Ландау // Воспоминания о Л.Д. Ландау. — М.: Наука, 1988.— 352 с.
2. Тарасевич Ю. Ю., Пономарева И. С. MATLAB в курсе численных методов и компьютерного моделирования // Тр. Всероссийской науч. конф. «Проектирование научных и инженерных приложений в среде MATLAB». Ч.6. MATLAB в Интернете и образовании.— М.: ИПУ РАН, 2002.— С.798–801.
3. Тарасевич Ю. Ю. Математическое и компьютерное моделирование. Вводный курс: Учебное пособие. 3-е издание, испр.— М.: Едиториал УРСС, 2003.— 144 с.
4. Ризниченко Г. Ю. Математические модели в биофизике и экологии.— Москва–Ижевск: Институт компьютерных исследований, 2003.— 184 с.

УДК 004

## **КОМПЛЕКС ВИРТУАЛЬНЫХ ЛАБОРАТОРНЫХ РАБОТ ПО ЭЛЕКТРОСНАБЖЕНИЮ И ЭНЕРГЕТИЧЕСКОЙ ЭЛЕКТРОНИКЕ**

*Хохлов Ю. И., Гельман М. В., Преображенский К. А., Федюнин К. П.  
Южно-Уральский Государственный Университет, Челябинск,  
e-mail: fedyunink@mail.ru*

У современных систем электроснабжения (СЭС) имеется большое количество потребителей, включаемых через вентильные преобразователи электрической энергии. К таким устройствам относятся преобразователи самой разной мощности от десятков мегаватт, питающих серии электролизных ванн и привода прокатных станов, до десятков ватт, питающих компьютеры и телевизоры. Это делает необходимым достаточно глубокое изучение студентами устройств энергетической электроники и их взаимодействия с питающей сетью, нагрузкой и друг с другом.

Обучение наиболее эффективно, если студенты изучают реальные объекты. Наблюдающаяся тенденция все большей интеграции элементов полупроводниковых преобразователей энергии затрудняет изучение студентами электромагнитных процессов, происходящих в преобразователях, из-за недоступности нужных точек схем. Разрабатываемые специализированные учебные лабораторные стенды также весьма дороги. Приобретение полного комплекта существующих устройств преобразовательной техники вызывает затруднения из-за ограниченных возможностей финансирования ВУЗов. При существующих трудностях становится проблематичной постановка новых лабораторных работ по новым направлениям преобразовательной техники и электроснабжения, особенно, если требуется фронтальное проведение работ. Вместе с тем, нельзя отрывать студентов от изучения материальных объектов, поэтому предлагается сочетать изучение физических моделей и компьютерное моделирование.

Из существующих пакетов прикладных программ моделирования электронных и электрических схем, одним из наиболее перспективным представляется пакет программ MATLAB 6, а именно моделирование в среде Simulink [1,2]. Пакет программ снабжен достаточным количеством примеров преобразователей, что позволяет ускорить создание новых моделей. Элементы описываются достаточно простыми параметрами, имеющимися в справочниках. Легко создаются приборы для определения энергетических показателей, имеются осциллографы и графопостроители.

Комплекс лабораторных работ по энергетической электронике включает циклы лабораторных работ: «Исследование компенсированных и некомпенсированных [3] управляемых и неуправляемых выпрямителей, ведомых инверторов и реверсивных преобразователей», «Исследование автономных инверторов напряжения и обратимых преобразователей напряжения», «Исследование преобразователей переменного напряжения» и «Исследование преобразователей постоянного напряжения».

Ниже приведены примеры некоторых лабораторных работ.

На рис. 1 приведена схема модели реверсивного двухкомплектного преобразователя. Она содержит блоки преобразователя, выполненные по трехфазной мостовой схеме, и систему управления. В блок трансформатора вводятся параметры приведенного трансформатора. Блок нагрузочно-питающее устройство содержит активно-индуктивную нагрузку и регулирующую ЭДС. В модели имеется возможность наблюдения переходного процесса и установившегося режима. На осциллограф выводятся вторичные токи и напряжения трансформатора, ток и напряжение на вентиле, ток и напряжение на нагрузке.

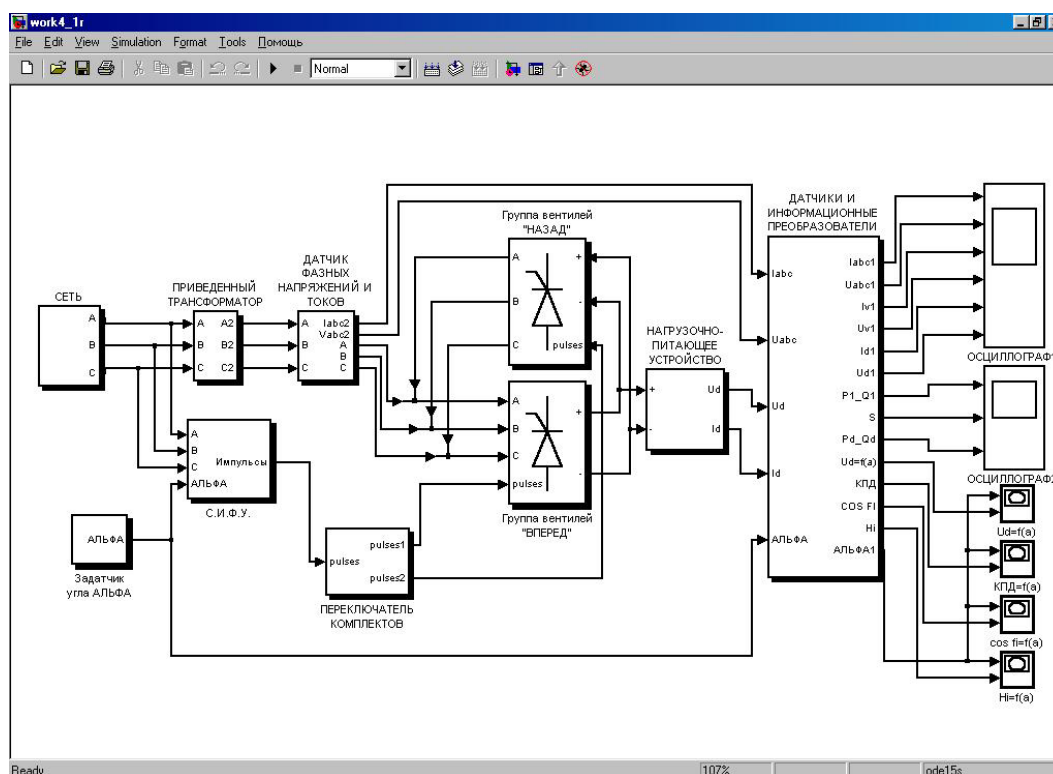


Рис. 1.

На рис. 2 приведены осциллограммы процессов в схеме в момент достижения углом управления комплекта «Вперед» 60 град. Модель предусматривает возможность получения на графопостроителе внешних, регулировочных и энергетических характеристик при медленном изменении ЭДС или угла управления. Регулирование тока производится изменением



ЭДС. Ток при изменении угла управления может поддерживаться постоянным за счет обратной связи по току.

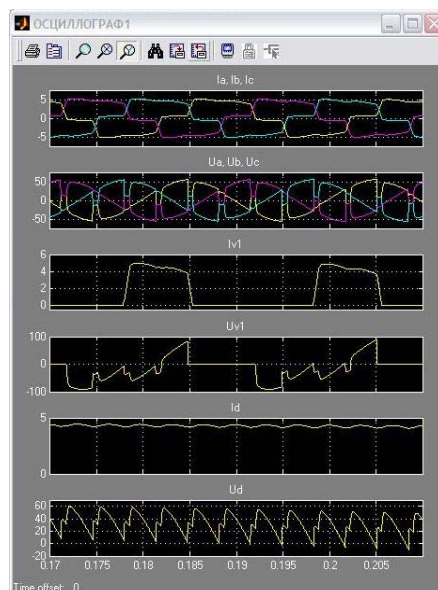


Рис. 2.

На рис.3 приведены регулировочные характеристики комплектов преобразователя, полученные с помощью графопостроителя. К сожалению, не удалось убрать вертикальные участки характеристик, вызванные переходным процессом при включении и переключении комплектов.

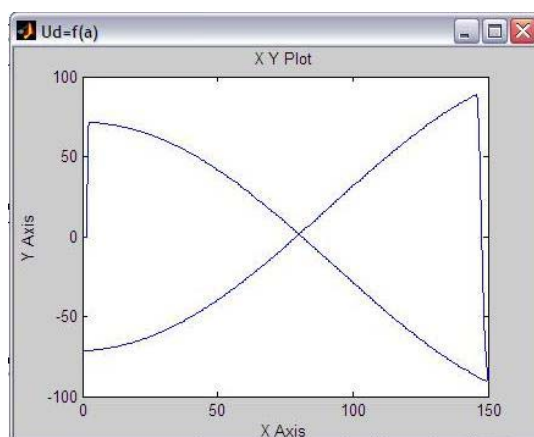


Рис. 3.

На рис. 4 приведена схема модели преобразователя переменного напряжения обеспечивающего плавный пуск асинхронного двигателя. Она содержит блок преобразователя, выполненный по встречно параллельной схеме, систему управления, блок питающей сети и блок нагрузки. Блок нагрузки содержит асинхронный двигатель. В модели имеется возможность наблюдения переходного процесса пуска, установившегося режима и оста-

новки. На осциллограф выводится фазный ток потребляемый из сети, линейные и фазные напряжения сети, напряжение на вентиле, заданный и электромагнитный моменты, скорость вращения двигателя, а также энергетические показатели:  $\cos \varphi$ , коэффициент несинусоидальности тока и коэффициент искажения синусоидальности питающего напряжения в точке присоединения преобразователя к сети.

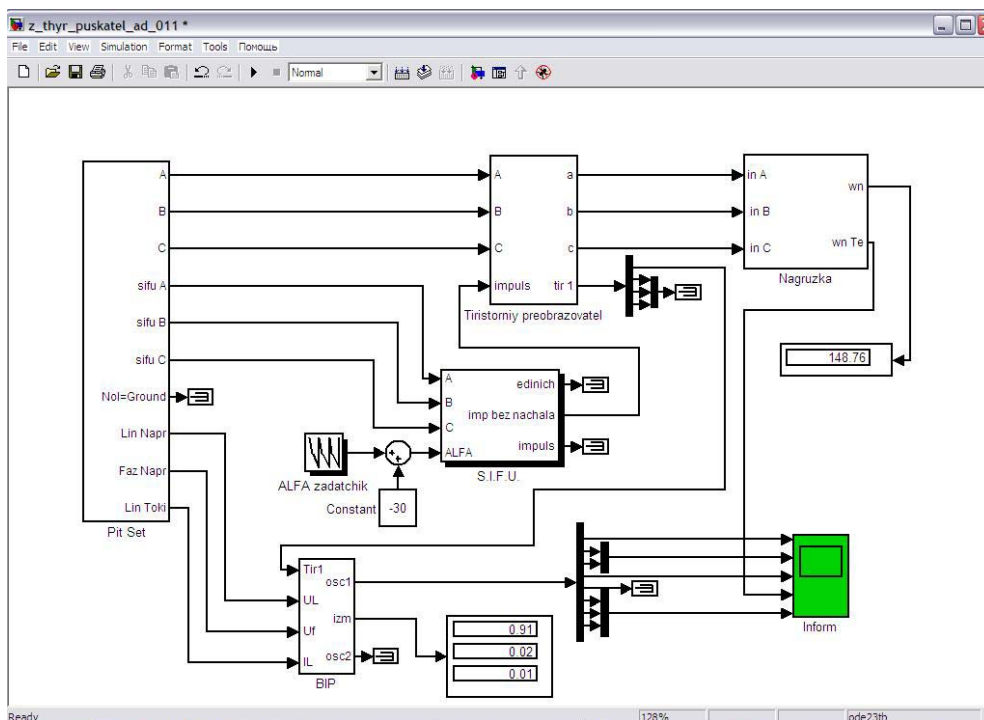


Рис. 4.

На (рис. 5) приведены осциллограммы процессов плавного пуска. Измерение энергетических показателей ведется в переходном режиме и определяется за период, поэтому эти показатели за первый период работы модели не могут быть правильно определены. На дисплеях отражаются действующие значения токов, энергетические показатели и скорость вращения двигателя.

Сравнение экспериментальных данных, полученных в лаборатории на некоторых физических объектах, и результатов моделирования позволяет студентам убедиться в адекватности моделей. В связи с широкими возможностями варьирования параметров и режимов виртуальное моделирование позволяет глубже изучить процессы.

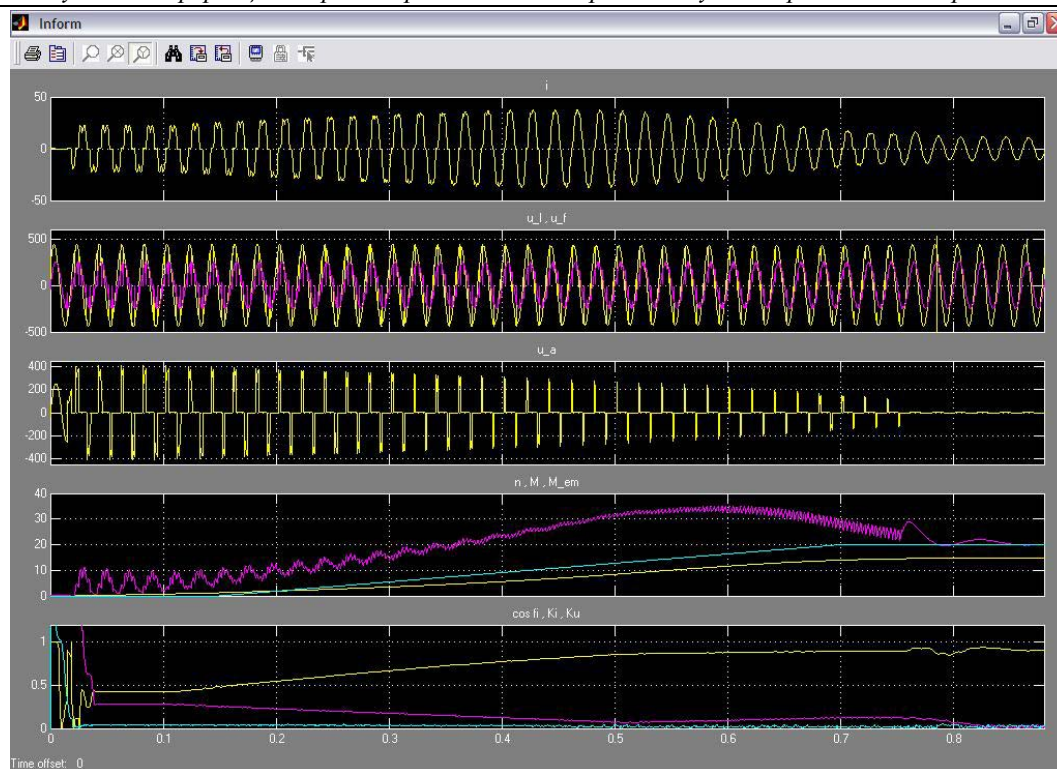


Рис. 7.

Не бывает систем без недостатков. В MATLAB плохо моделируются некоторые процессы. Трудно производить измерения при частотном регулировании, так как все измерители настраиваются на определенные частоты. Трудно моделировать различные сложные схемы соединения обмоток трансформатора.

### Литература

1. Гельман М. В., Хроменков М. А. Комплекс лабораторных работ по курсу «преобразовательная техника» на основе компьютерного моделирования // Электротехнические системы и комплексы: Межвузовский сборник научных трудов.— Магнитогорск: МГМА, 1996.— 200.— С.32.
2. Герман-Галкин С. Г. Силовая электроника: Лабораторные работы на ПК.— СПб.: КОРОНА принт, 2002.— 304.
3. Хохлов Ю. И. Компенсированные выпрямители с фильтрацией в коммутирующие конденсаторы нечетнократных гармоник токов преобразовательных блоков.— Челябинск: ЧГТУ, 1995.— 355 с.

УДК 001.007:378

## **ОРГАНИЗАЦИЯ ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКИ СТУДЕНТОВ С ИСПОЛЬЗОВАНИЕМ ВОЗМОЖНОСТЕЙ ПАКЕТА MATLAB**

*Чернецова Е. А.*

*Российский государственный гидрометеорологический  
университет, Санкт-Петербург,  
e-mail: mit@rshu.ru*

Кафедра «Морские информационные технологии» Российского государственного гидрометеорологического университета производит подготовку специалистов по двум специальностям: «Морские информационные системы и оборудование» и «Информационная безопасность телекоммуникационных систем». Для студентов второго курса этих специальностей предусмотрена летняя вычислительная практика длительностью один месяц, во время которой они должны приобрести навыки работы в пакете MATLAB, которые им будут необходимы в дальнейшем для обучения на старших курсах и в период написания дипломных работ.

Студентам предлагается тринадцать заданий, ориентированных как на общее знакомство с возможностями пакета MATLAB, так и на исследование временных рядов с использованием различных MATLAB toolbox. Задания включают в себя знакомство с простейшими математическими операциями и построение двумерных и трехмерных графиков, спектральную обработку временных рядов, построение двумерных и трехмерных гистограмм, аппроксимацию распределений, обработку временных рядов с использованием вейвлет-преобразований, создание удобного для пользователя интерфейса программы с использованием Callback-функций, а также создание в среде MATLAB исполняемых exe-файлов.

Особенностью заданий практики является наличие вариантов, позволяющих каждому студенту решать индивидуальную задачу в рамках темы каждого задания, а также наличие большого количества исследуемых временных рядов — записей реальных сигналов, полученных от различных датчиков и хранящихся в виде файлов в бинарных кодах и файлов в ASCII-кодах.

Рассмотрим предлагаемые студентам работы и порядок их выполнения.

### ***Задание №1***

*Цель работы:* приобретение навыков работы в рабочем окне пакета MATLAB: изучение способов формирования векторов и матриц, вычисле-

ние элементарных математических и тригонометрических функций, вывод двумерных и трехмерных графиков функций.

*Задание на выполнение работы:*

1. Ввести в рабочем окне матрицу чисел  $A$ , выделить из нее подматрицу  $B$  по определенному принципу, умножить матрицу  $B$  на матрицу  $C$  с точкой и без точки.

2. Разделить матрицу  $B$  на матрицу  $C$  левым делением с точкой и без точки.

3. Разделить матрицу  $B$  на матрицу  $C$  правым делением с точкой и без точки.

4. Транспонировать матрицу  $A$ .

5. Построить с помощью функции `plot` вектор, состоящий из всех строк третьего столбца матрицы  $A$ .

6. Построить трехмерный график матрицы  $A$  в зависимости от номера элемента по строкам и столбцам с использованием команд `mesh` и `surf` с использованием различной цветовой палитры и с возможностью поворачивать изображение под разными ракурсами.

7. Сохранить переменные  $A, B, C, D$  в `mat`-файле.

8. Очистить рабочую область. Очистить экран. Загрузить сохраненные переменные в рабочую область. Вывести список переменных.

### ***Задание № 2***

*Цель работы:* приобретение навыков работы с `m`-файлами; создание программ-функций, работа с циклами и условными операторами.

*Задание на выполнение:*

1. С помощью команды `solve`, сформированной в командной строке, решить систему уравнений (вариант задается преподавателем).

2. Найти пределы функций (вариант задается преподавателем)

3. Используя конструкции условных операторов и операторов цикла, сформировать программу-функцию обработки матрицы (вариант обработки задается преподавателем, например, преобразовать матрицу  $A$  в матрицу  $B$ , у которой на главной диагонали стоят единицы).

### ***Задание № 3***

*Цель работы:* приобретение навыков обработки и графического представления файлов данных, записанных в ASCII-кодах.

*Задание на выполнение работы:*

Загрузить ASCII-файл в рабочую область, вывести график сигнала, нанести сетку, включить масштаб и выбрать окно для анализа в отсчетах. Вывести график выбранного столбца данных с использованием масштабирования данных, с подписями заголовка и названий осей, нанести сетку. Оформить в виде процедуры-функции вычисление огибающей и спектра для выбранного отрезка временного ряда. Вывести графики огибающей и

спектра временного ряда. Сохранить данные о спектре в файле формата Lotus, открыть его в Excel и построить график спектра с использованием инструментария Excel.

#### **Задание № 4**

*Цель работы:* Обработка временных рядов и графическое представление статистических характеристик.

*Задание на выполнение работы:*

Загрузить в рабочую область файл экспериментальных данных и построить гистограмму мгновенных значений. Построить плотность распределения вероятностей известного закона распределения с использованием команд из MATLAB Statistic Toolbox и аппроксимировать ею экспериментальную ПРВ по критерию хи-квадрат, добившись приемлемой точности аппроксимации. Вывести график исследуемой и образцовой ПРВ.

#### **Задание № 5**

*Цель работы:* исследование временных рядов с помощью вейвлет-преобразования.

*Задание на выполнение работы:* Исследовать заданный временной ряд с помощью графического интерфейса 1-D и Packet 1-D вейвлет-преобразования, а также написать программу для извлечения заданных коэффициентов детализации.

#### **Задание № 6**

*Цель работы:* приобретение навыков работы с файлами данных, записанных в различных форматах .

*Задание на выполнение работы:*

Написать программу-функцию по открытию экспериментально полученного файла. Произвести заданную преподавателем спектральную или временную обработку временного ряда и сохранить результаты в excel-совместимом файле с расширением wk1.

#### **Задание № 7**

*Цель работы:* знакомство со script-файлами и организация диалогового режима с пользователем.

*Задание на выполнение работы:*

1. Создать файл-сценарий открытия файла, записанного в ASCII-кодах и файл-сценарий открытия экспериментального файла, записанного в бинарных кодах с использованием диалогового режима ввода пути к файлу и выдачей сообщения об ошибке в случае ввода неправильного пути.

2. Создать сценарий открытия файла, записанного в ASCII-кодах и сценарий открытия файла, записанного в бинарных кодах с использованием поиска через проводник.

### Задание № 8

*Цель работы:* создание удобного интерфейса пользователя для выполнения файла-сценария.

*Задание на выполнение работы:*

Написать программу интерфейса, выводящую на экран фигуру с управляющими кнопками (рис. 1).

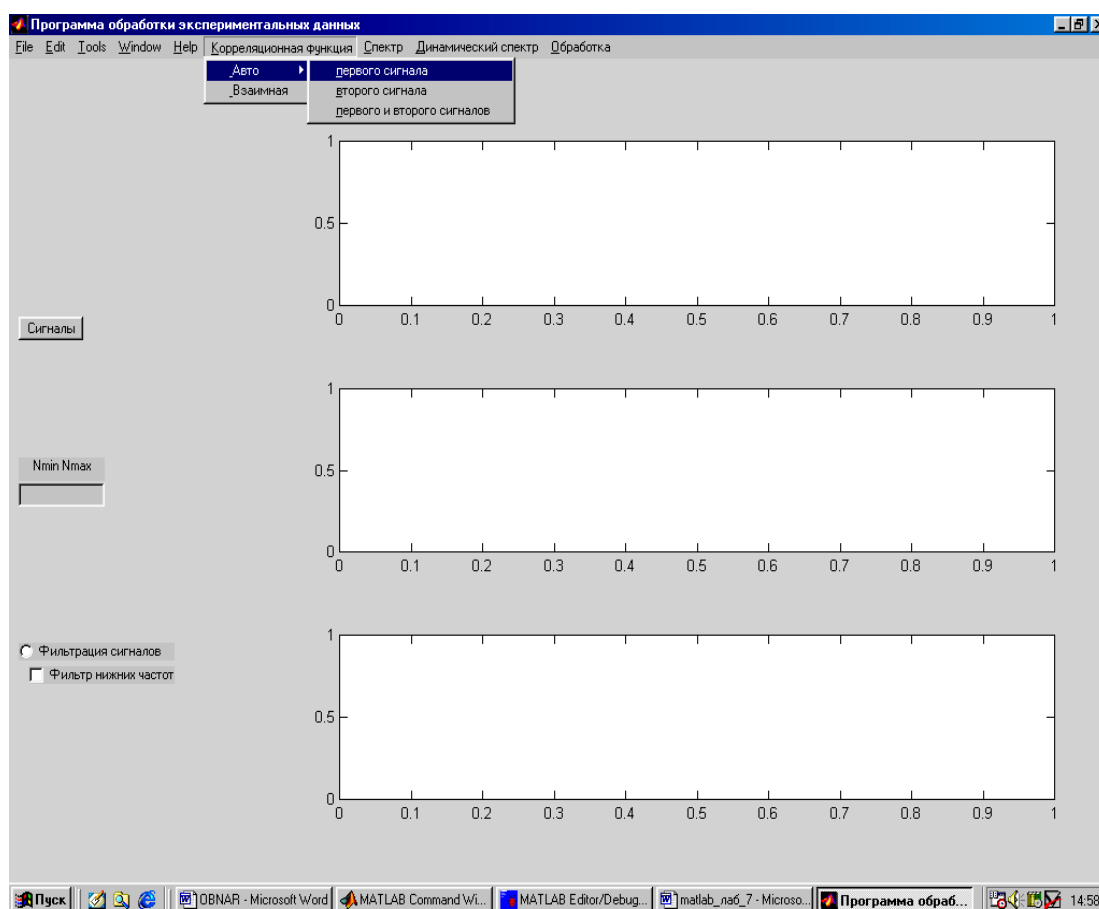


Рис. 1.

### Задание № 9

*Цель работы:* написание callback-подпрограммы для программы, создающей интерфейс пользователя.

*Задание на выполнение работы:*

Создать окно интерфейса, показанное на рис. 2 с возможностью выбора открытия бинарного файла или файла в ASCII-кодах, написать callback подпрограмму для этого интерфейса с возможностью выбора анализируемого фрагмента сигнала путем задания величины в окне масштаба.

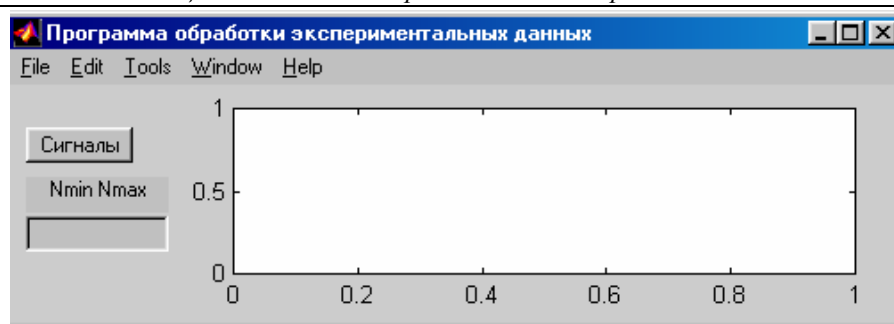


Рис. 2.

**Задание № 10**

*Цель работы:* создание в среде MATLAB исполняемых файлов.

*Задание на выполнение работы:*

Создать в среде MATLAB исполняемую программу, позволяющую открывать бинарные файлы .

**Задание № 11**

*Цель работы:* исследование временных характеристик сигнала с использованием графического интерфейса.

*Задание на выполнение работы:*

Создать графический интерфейс (рис. 3) и CallBack — программу к нему для расчета временных характеристик заданного сигнала.

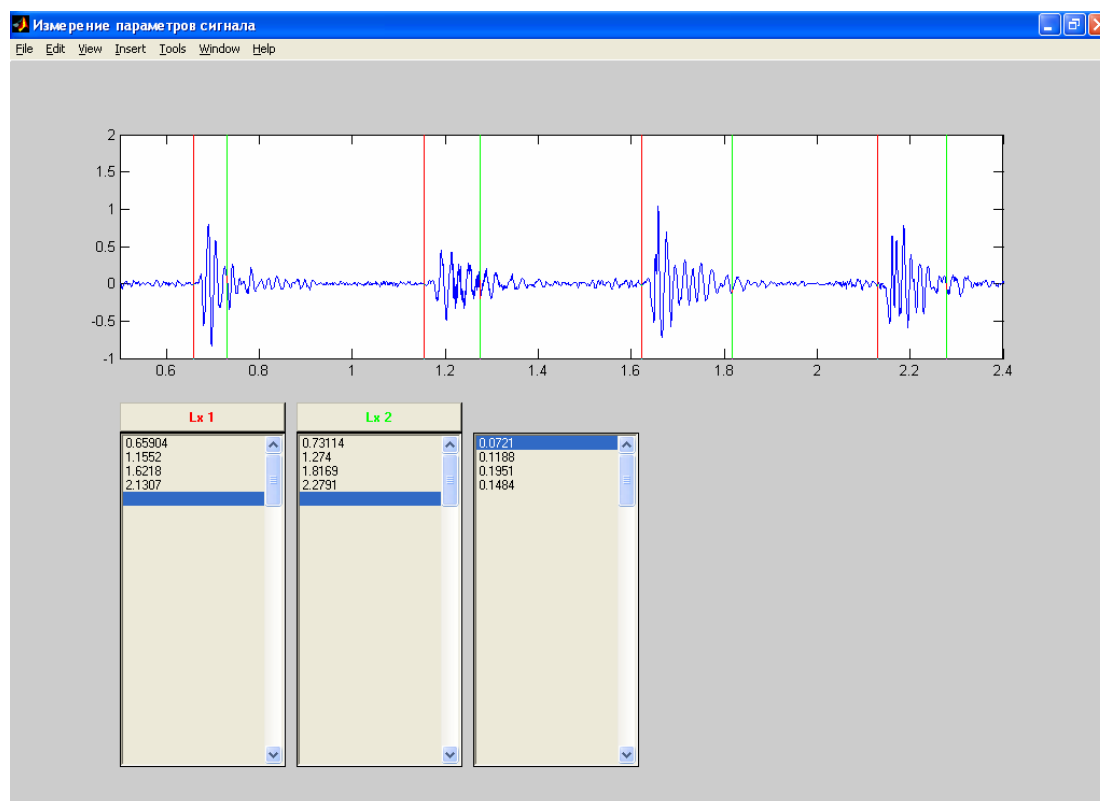


Рис. 3.



***Задание № 12***

*Цель работы:* построение гистограммы двух временных рядов в трехмерном пространстве.

*Задание на выполнение работы:*

Написать программу построения гистограммы двух временных рядов в трехмерном пространстве и определить параметры полученного распределения вероятностей.

***Задание № 13***

*Цель работы:* знакомство с применением теории информации для анализа временных рядов.

*Задание на выполнение работы:*

Рассчитать заданные преподавателем информационные расстояния для двух временных рядов.

УДК 378.6(47+57-89):621.43

## **РАЗВИТИЕ И ПРИМЕНЕНИЕ НОВЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В УЧЕБНОЙ И ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ С ИСПОЛЬЗОВАНИЕМ ПАКЕТА MATLAB И ЕГО ПРИЛОЖЕНИЙ**

*Шатров М. Г., Черняк Б. Я.*

*Московский автомобильно-дорожный институт  
(государственный технический университет), Москва,  
e-mail: dvs@madi.ru*

Кафедра «Теплотехника и автотракторные двигатели» МАДИ-ГТУ (ТиАТД) на протяжении длительного времени ведет активную разработку методов использования интеллектуальных технологий в НИР и учебном процессе.

В результате выполненных работ к настоящему времени создан и широко реализован в учебном процессе учебник-комплекс «Двигатели внутреннего сгорания», состоящий из учебника в трех томах и Интегрированный обучающий комплекс «ДВС» (ИОК «ДВС»), который удостоен премии Правительства Российской Федерации 1999 года в области науки и техники.

Разработанный ИОК «ДВС» обеспечивает изучение студентами курсов ДВС уже в более чем 100 вузах различных регионов России, стран СНГ, Европы, Латинской Америки, Азии и Африки. Он позволяет обеспечить современный уровень энергоэкологической подготовки студентов по направлениям: 551400 «Наземные транспортные системы», 552700 «Энергомашиностроение», 551800 «Технологические машины и оборудование», 552100 «Эксплуатация транспортных средств». Компоненты ИОК «ДВС» могут успешно использоваться при дипломном проектировании как для специальности 101200 (ДВС), так и для других специальностей с учетом их специфики при разработке и эксплуатации ДВС на транспортных средствах и дорожно-строительных машинах. Ряд элементов ИОК ДВС может быть использован непосредственно или при их адаптации при подготовке специалистов водного и морского транспорта, авиации, лесного и сельского хозяйства.

В настоящее время разработана испанская версия ИОК «ДВС», которая в 2004 году выходит на рынок Латинской Америки.

Созданная методика подготовки дидактических средств для учебного процесса позволяет наращивать объем формализованной и структурированной информации для обеспечения полного жизненного цикла поршневого ДВС.

Наиболее важным сейчас является создание учебно-методических комплексов (УМК), органично увязывающих в систему все компоненты учебного процесса (лекций, лабораторных работ, проектирования, средств контроля качества, самостоятельной работы) и использующих как традиционные формы организации учебного процесса, дидактические средства (учебники, учебные и методические пособия, лабораторную базу), так и средства, базирующиеся на современных информационных и коммуникационных технологиях, обеспечивающих самостоятельную работу обучающегося.

В качестве основы УМК используется единое информационное поле предметной области ДВС. Первоочередными задачами при его создании являются разработка состава и структуры поля, методов его наполнения и пополнения информацией, а также методов работы в информационном поле, обеспечивающих высокое качество усвоения информации при ограниченном времени обучения. В связи с постоянным увеличением объема информации, развитием знаний об изучаемой предметной области на первый план выходит отработка у обучающегося технологий усвоения и получение новой информации.

Основу развития НИР и учебного процесса кафедры составляет разработка методик, технологий и программного обеспечения для комплексного использования новых методов решения инженерных задач при изучении дисциплин специальности 101200 (ДВС). Наиболее перспективными направления работ здесь в настоящее время являются:

- моделирование динамики рабочих процессов ДВС,
- вопросы прочности, надежности, тепловой и динамической напряженности деталей в динамических процессах,
- динамические процессы в системах топливоподачи,
- автоматическое управление,
- электронное управление процессами ДВС,
- шум и вибрации в ДВС,
- методы моделирования и обработки экспериментальных данных.

Специфика современного этапа состоит в том, что интеграция науки и обучения настолько велика, что попытки их разделения при изучении сложных технических объектов, их исследовании и доводке становится тормозом в построении эффективных систем обучения специалистов. Более тщательная отработка методики и технология исследования, которую требует учебный процесс, позволяет более эффективно использовать разработанную систему в научных исследованиях.

С целью повышения системности в организации данной работы естественно мы постоянно отслеживаем состояние инструментальных средств, необходимых для обеспечения этого. Следует отметить, что информационные и коммуникационные технологии на данном этапе развиваются стремительно. В настоящее время бурно развиваются Интернет-

технологии, языки программирования. Можно отметить новый уровень практической реализации принципов CALS-технологий. Активно развиваются технологии и пакеты для различной научной и инженерной деятельности, которые позволяют охватить значительную часть проблематики современных поршневых ДВС. К ним следует отнести:

- стохастическое имитационное моделирование с использованием матричных моделей для моделирования процессов в двигателе в реальном масштабе времени;
- новейшие оптимизационных технологий таких, как метод непрямой стохастической оптимизации на основе самоорганизации, метод конфигураций и др.;
- искусственные нейронные сети для построения систем управления и идентификации моделей;
- элементы нечеткой логики при решении оптимизационных задач с качественными критериями и др.

Пакет MATLAB и его приложения в большой степени отвечает этим требованиям.

Вопросам моделирования в ДВС на кафедре всегда уделялось большое внимание. Использование здесь MATLAB и его приложений открыло новые возможности как в НИР, так и при создании соответствующих учебных пособий и материалов, а также в методах организации и методиках обучения.

Кафедрой ТиАТД МАДИ совместно с НПП ЭЛКАР создан научно — учебный центр для развития методов использования MATLAB и его приложений при совершенствовании систем управления ДВС и в учебном процессе. С этой целью приобретены лицензионные версии MATLAB. На базе лаборатории НПП ЭЛКАР студенты специальности 101200 (Двигатели внутреннего сгорания) проходят практику и используют MATLAB при подготовке магистерских и дипломных работ.

К сожалению, нам пока не удалось ввести в учебный процесс систематическое изучение пакета MATLAB всеми студентами специальности (хотя, понимая возможности пакета, такой план имеется и предполагается поэтапное введение курса в учебный план вначале факультативного, а затем обязательного). Сейчас решаются вопросы формирования комплекса дисциплин, состоящего из общего курса, описывающего принципы построения и возможности пакета MATLAB (по-видимому, он должен быть многоуровневый: от ознакомительного до, в зависимости от потребностей, более углубленного с рассмотрением отдельных возможностей) и, в нашем случае, в виде отдельных курсов или разделов других специальных дисциплин для приложений MATLAB в инженерной и научной практике предметной области ДВС. Он должен быть гармоничным, эффективно функционирующим и развиваемым комплексом. Сложность вопроса в том, что первый курс должна читать кафедра математики, а вопросы приложе-

ний — специальные кафедры. Требуемый уровень и объем общего курса должен удовлетворять разноплановым потребностям специальных кафедр вуза, что зачастую сложно сбалансировать. И здесь требуется подготовка синтетического учебного плана, позволяющего настраиваться под задачи того или другого вуза.

В настоящее время на специальных занятиях все студенты, слушающие на кафедре курсы по МСУ, знакомятся с возможностями пакета и ряд студентов с помощью преподавателей самостоятельно овладевают им.

На кафедре накоплен практический опыт использования MATLAB: завершены две кандидатские диссертации, защищены магистерская диссертация, три дипломных проекта, ряд бакалаврских работ и учебно-исследовательских проектов.

Опыт работы аспирантов, а также студентов в рамках выполнения ими исследовательских и дипломных проектов с использованием MATLAB, показал, что они легко и с большим интересом овладевают этими новыми методами решения инженерных задач. При этом их кругозор очень быстро расширяется, а производительность резко увеличивается. Очевидно, что ближайшей перспективе, владение интеллектуальными технологиями станет обязательным для инженеров.

На базе MATLAB на кафедре сейчас создаются ряд новых лабораторных работ в первую очередь по дисциплинам «Теория автоматического управления», «Электронные системы управления ДВС», «Управление рабочими процессами двигателей».

Мы предлагаем создать консультационный или учебный центр и проводить постоянно действующий семинар для распространения наработанных методов и решений задач в области ДВС с использованием MATLAB среди родственных кафедр.

Кафедра заинтересована в расширении сотрудничества со всеми организациями в области совершенствования использования MATLAB в учебном процессе.

MATLAB весьма совершенный инструмент. Мы уверены, что изучение и использование современных интеллектуальных средства для расширения возможностей инженерных исследований и разработок должно стать обязательным элементом при подготовке инженеров-двигателистов.

### ***Литература***

1. Двигатели внутреннего сгорания. В 3 кн. Кн. 1: Теория рабочих процессов: Учеб. / В. Н. Луканин, К. А. Морозов, А. С. Хачиян и др.; под ред. В. Н. Луканина.— М.: Высш.шк., 1995.— 368 с.
2. Двигатели внутреннего сгорания. В 3 кн. Кн. 2: Динамика и конструирование: Учеб. / В. Н. Луканин, К. А. Морозов, А. С. Хачиян и др.; под ред. В. Н. Луканина.— М.: Высш.шк., 1995.— 319 с.

3. Двигатели внутреннего сгорания. В 3 кн. Кн. 3: Компьютерный практикум: Учеб. / В. Н. Луканин, К. А. Морозов, А. С. Хачиян и др.; под ред. В. Н. Луканина.— М.: Высш.шк., 1995.— 256 с.