

ИНФОРМАТИКА

УДК 658.012.011.56

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИНТЕГРАЦИИ ИМИТАЦИОННОЙ
МОДЕЛИ С КОМПЛЕКСНОЙ ИНФОРМАЦИОННОЙ СИСТЕМОЙ**

А.И. ЯКИМОВ, К.В. ЗАХАРЧЕНКОВ

*Белорусско-Российский университет
пр. Мира, 43, Могилев, 212005, Беларусь**Поступила в редакцию 14 марта 2008*

Представлено описание программно-технологического комплекса имитации сложных систем *BelSim*, схемы постановки имитационных экспериментов. Для интеграции имитационной модели с комплексной информационной системой предприятия разработано программное приложение *Data Integrator*, показаны особенности интеграции на практическом примере решения задачи распределения производственной нагрузки.

Ключевые слова: имитационное моделирование, информационная система, программный комплекс, интеграция.

Введение

Одной из функций ERP-систем управления является моделирование. Для ее выполнения на основе комплексных информационных систем (КИС), используя единое хранилище данных, внедряются системы поддержки принятия решений, использующие аналитические модели, например, *SAP BW*, модуль *Контроллинг* в КИС "Галактика". Однако для повышения эффективности систем поддержки принятия решений в условиях КИС необходимо использовать имитационные модели из-за высокого уровня детализации процессов на предприятии [1].

Базовой предпосылкой проведенных исследований является предположение о возможности эффективного использования метода имитационного моделирования сложных систем при комплексном применении современных информационных технологий для построения имитационных моделей (ИМ) производственно-экономической деятельности промышленного предприятия и решения задач прогнозирования и оптимизации [2].

Имитационное моделирование в ПТКИ BELSIM

Программно-технологический комплекс имитации сложных систем (ПТКИ) *BelSim* позволяет реализовать метод имитационного моделирования производственно-экономической деятельности промышленных предприятий. В состав комплекса входит следующее программное обеспечение: для построения функциональной модели системы на основе технологии *IDEFO*; интегрированная среда разработки приложений на языке *C++*; система имитационного моделирования *PSTL* (*Process Simulation Template Library*); для планирования, проведения и обработки результатов имитационных экспериментов; решения оптимизационных задач; анализа и отображения данных; интеграции имитационных моделей в КИС; организации распределенных вычислений. При разработке сложных моделей рекомендуется использовать среду разработки,

обладающую развитыми инструментальными средствами написания и отладки программ (например, *Microsoft Visual C++.NET*) [1].

Особенностью реализации подсистемы планирования, проведения и обработки результатов имитационных экспериментов является организация обмена данными между элементами подсистемы на основе иерархических структур, физически представленных в виде файлов в формате *XML*, доступ к которым реализован через стандартный интерфейс *XML DOM Level 3* с использованием библиотеки *MSXML 4.0* [3]. Блок ввода/вывода информации реализован в виде отдельного приложения *Experiment Designer* на основе *Microsoft .NET Framework 1.1*. Блок создания плана эксперимента, блок формирования данных для отдельных опытов и блок предварительного статистического анализа реализованы в виде макросов *DesignOfExperiment* и *ExperimentData* в специализированном пакете статистического анализа *STATISTICA 6.0* (рис. 1).

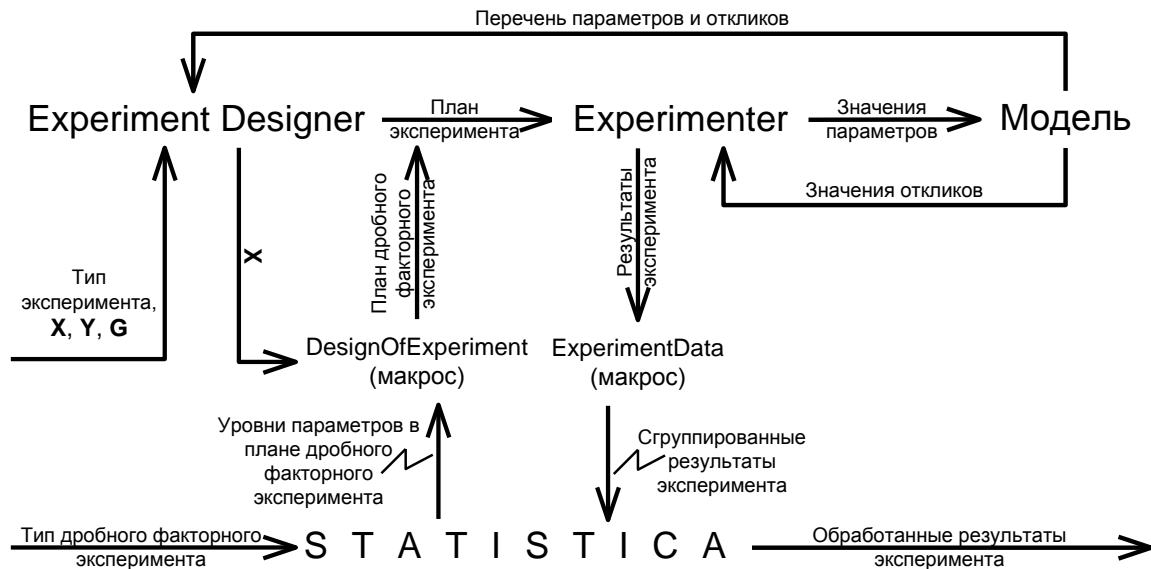


Рис. 1. Схема постановки имитационных экспериментов в ПТКИ *BelSim*

ПТКИ *BelSim* обладает методами и средствами проведения структурного анализа системы и протекающих в ней процессов; способностью одновременного моделирования материальных, денежных и информационных процессов. Имеется возможность интеграции модели в информационную систему предприятия с целью получения исходных данных для моделирования и использования модели в составе системы управления.

В состав *BelSim* входит базовая ИМ промышленного предприятия, в основу построения которой положен стандарт MRPII/ERP [4]. Для решения конкретных задач соответствующий компонент ИМ подвергается требуемой детализации. В случае если имеется достаточно статистических данных, реальный процесс можно аппроксимировать функциональными зависимостями.

Интеграция ПТКИ *BelSim* с комплексной информационной системой

Для интеграции ПТКИ *BelSim* с КИС выполняется последовательность действий, определяемая следующими этапами (рис. 2).

Этап 1. Настройка параметров подключения к базе данных.

Шаг 1.1. Выбор пути к базе данных, из которой необходимо получить данные для исследуемой модели.

Шаг 1.2. Выбор файла, в котором хранится структура *XML*-файла данных модели.

Шаг 1.3. Формирование запроса к базе данных с указанием названий таблиц и полей, в которых находятся требуемые данные.

Этап 2. Подключение к базе данных предприятия или КИС (например, "ИС Предприятие") и выбор требуемых данных с применением технологии *ADO*, используя библиотеку *msado15.dll*, входящую в дистрибутив операционной системы *Windows XP Professional*.

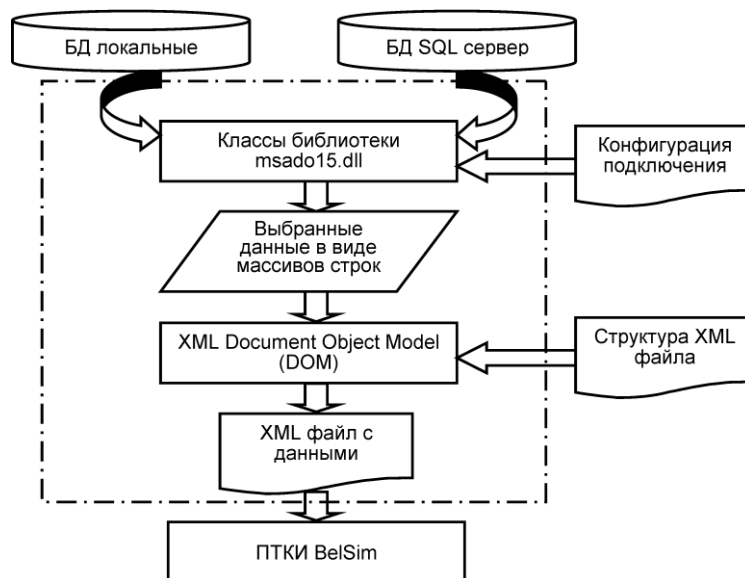


Рис. 2. Логическая структура интеграции ПТКИ BelSim в КИС

Этап 3. Запись полученной информации в *XML*-файл данных модели (после выбора данных из КИС предприятия их следует представить в виде *XML*-файла, который ПТКИ *BelSim* может использовать для проведения имитационных экспериментов). Интерфейсом доступа к содержимому *XML*-документов является объектная модель документа *DOM* (*Document Object Model*), представляющая внутреннюю структуру документа в виде совокупности объектов [5].

Программное приложение *Data Integrator* (рис. 3) считывает параметры подключения к базе данных из конфигурационного файла *Settings.cfg*, выполняет подключение к базам данных и выбор требуемых данных из таблиц, используя объекты классов библиотеки *msado15.dll*. Используя *DOM* и файл со структурой будущего документа, составляется конечный *XML*-файл, который передается в *Experiment Designer* ПТКИ *BelSim* (на рис. 1 *XML*-файл условно не показан).

Основным классом, ответственным за сбор и конвертирование данных в приложении *Data Integrator*, является *DBF2XML* со следующими атрибутами: *NumAllParameters* — задает общее число параметров файла *Settings.cfg*, *BelsimPath* — определяет путь к директории, в которой установлен ПТКИ *BelSim*, *OnecPath* — путь к директории, в которой хранятся базы данных; *ModelStructureXML* — имя *XML*-файла, загружаемого в качестве шаблона выходного документа; *TablesAndColumns* — указатель на динамический массив, содержащий названия таблиц и полей, необходимых для импорта данных; *SQLServer* — имя сервера, на котором хранятся базы данных (если для хранения данных используется СУБД *SQL Server*, рис. 2)

Ядром приложения *Data Integrator* является метод *convert()* класса *DBF2XML*, обеспечивающий сбор и конвертирование данных по следующему алгоритму: создать объект класса *CRealXMLMData*; загрузить структуру *XML*-файла; создать объекты *Connection* и *Recordset*, используя *COM*; установить соединение с базой данных; для каждой таблицы и колонки сформировать и выполнить запрос к базе данных, создать динамический массив строк, преобразовать к типу *CString* и копировать записи из запроса в массив строк, копировать имена и значения параметров из массива строк в *XML*-файл данных модели; закрыть соединение с базой данных. Используемый в реализации алгоритма класс *Connection* предназначен для хранения информации о подключении к источнику данных, такой как имя, местоположение данных, имя пользователя и пароль, имя провайдера *OLE DB* и т.д. Основные функции класса *Connection* реализуют открытие источника данных, выполнение запроса к базе данных и возврат результата, закрытие соединения с источником данных.

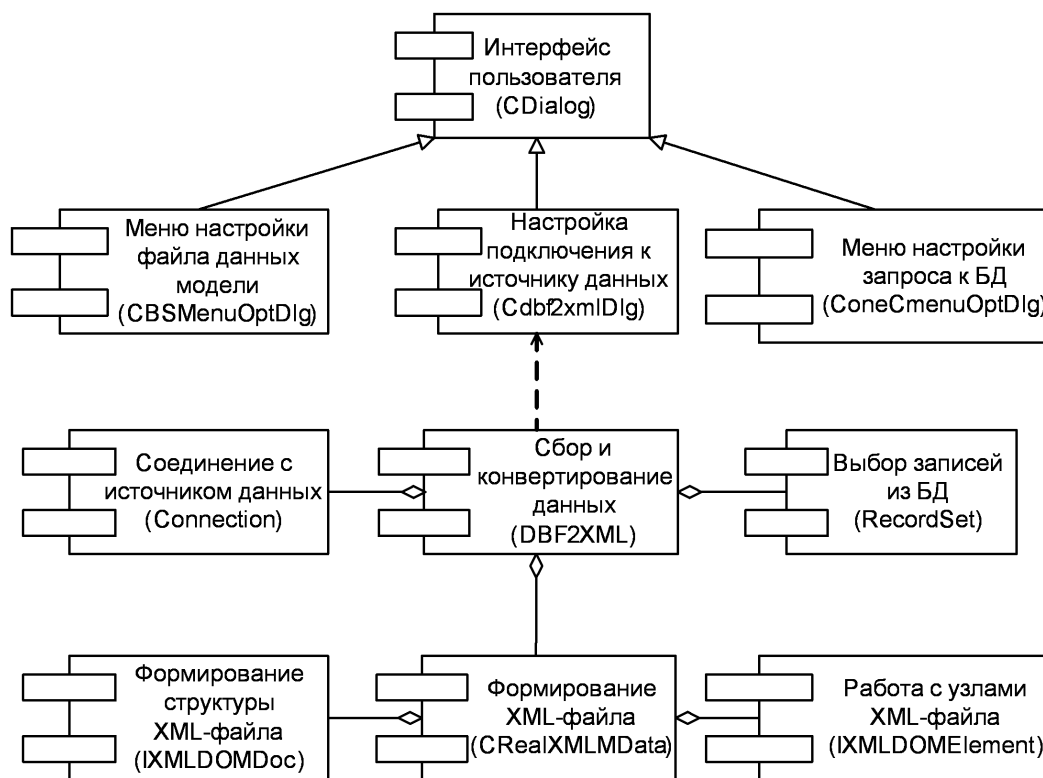


Рис. 3. Диаграмма классов приложения *Data Integrator*

Класс *Recordset* содержит реализацию методов, позволяющих пользователю обновлять данные в базе данных и перемещать указатель текущей записи по результирующему набору. Также этот объект содержит поля, хранящие число записей в наборе, тип курсора, тип блокировки и т.д. Функции класса *Recordset*: *MoveFirst* — передвигает курсор на первую запись; *MoveNext* — передвигает курсор на следующую запись; *RecordCount* — возвращает количество записей; *adoEOF* — возвращает true, если курсор находится на первой записи; *GetCollect* — считывает текущее значение записи.

Класс *CRealXMLModelData* (рис. 3) обеспечивает доступ к классам библиотеки *MSXML* на основе COM-технологии с использованием директивы `#import <msxml4.dll>`. При этом используются функции класса *IXMLDOMDoc* и функции класса *IXMLDOMElement*. Функции класса *IXMLDOMDoc* следующие: *AppendChild* — добавляет новый дочерний узел как последний (в конец списка) дочерний узел данного узла; *CreateElement* — создает узел с заданным именем; *Load* — загружает XML-документ по указанному местоположению; *Save* — сохраняет XML-документ по заданному местоположению. Основные функции класса *IXMLDOMElement*: *SetAttribute* — присваивает значение указанному атрибуту; *findVariable* — возвращает первый найденный узел переменной; *setAttributeNode* — добавляет или изменяет указанный атрибут; *findVariableArray* — возвращает первый найденный узел переменной массива.

Функции классов *CDialog*, *CBSMenuOptDlg*, *Cdbf2xmlDlg*, *COneCmenuOptDlg* (рис. 3) обеспечивают реализацию пользовательского интерфейса.

Технология решения практической задачи на основе интеграции модели с комплексной информационной системой

На крупных химических предприятиях расход средств по основным статьям, определяющим себестоимость продукта (расход сырья, расход энергетических ресурсов, амортизационные отчисления), с увеличением мощности производства при сохранении технологической схемы и режима проведения основных процессов имеет тенденцию к сокращению. В то время как расход сырья на стадии химического превращения практически не зависит от размера про-

изводственной аппаратуры, относительные потери сырья и готового продукта (например, ДМТ на заводе органического синтеза (ЗОС)), обратно пропорциональные величине перерабатываемых потоков на других стадиях технологической схемы, уменьшаются с увеличением мощности производства почти прямолинейно, приводя к снижению удельного расхода сырья на единицу готового продукта. Если производство многопоточное, то для заданной производственной программы ставится задача определения оптимальной загрузки цехов предприятия [6, 7].

Для исследования непрерывных производственных процессов в ПТКИ *BelSim* разработана имитационная модель [8], предварительные исследования которой с использованием накопленной в КИС базы со статистическими данными позволили построить рабочую модель с применением регрессионных зависимостей. Особенности использования такой модели в реальных условиях производства представлены ниже.

Комплексная информационная система "1С: Предприятие 8.0" может использоваться внешними приложениями в качестве *OLE*-сервера. Для запуска "1С: Предприятие 8.0" в качестве *OLE*-сервера, например из *MS Excel*, выполняется следующая последовательность действий: создается *OLE*-объект с идентификатором "v8.Application"; выполняется инициализация системы "1С: Предприятие 8.0" методом *Connect*; вызываются свойства и методы системы "1С: Предприятие 8.0" как *OLE*-сервера.

Выбрать необходимые данные из справочника можно при помощи запроса. В системе "1С: Предприятие 8.0" использование запросов является основным способом получения требуемых данных. Для выполнения запроса к данным *OLE*-базы достаточно с помощью метода *NewObject* создать объект "Запрос" и далее работать с ним, используя стандартные методы. Для получения данных при помощи "Запроса" из справочников, содержащих информацию о ценах на материалы и энергоресурсы, используется программа на языке *Visual Basic*.

После получения из системы "1С: Предприятие 8.0" данные сохраняются на отдельном листе *MS Excel* в строго определенных ячейках. Этого можно добиться, задавая различные условия при запросе и используя сортировку. Например, при запросе данных о ценах на материалы, они сортируются по коду и, несмотря на то, что запрос выбирает все поля, в ячейки можно вставлять значения любых атрибутов строки запроса, используя доступ к ним как к элементам объекта "Запрос". На данные о ценах и нормах расхода материалов устанавливаются ссылки из модели. Таким образом, обеспечивается автоматизированный ввод исходных данных в модель [9].

Технология моделирования производства ДМТ включает следующие основные этапы.

Этап 1. Импорт данных.

Шаг 1.1. В модель импортируются данные о ценах и нормах расхода основного сырья, вспомогательных материалов, и также о ценах на энергоресурсы из информационной системы "1С: Предприятие".

Шаг 1.2. Данные о расходе энергоресурсов в зависимости от выработки продукции сохраняются в специализированной информационной системе на базе *MySQL* в виде статистической информации, полученной снятием показаний контрольно-измерительных приборов.

Этап 2. Обработка статистических данных.

Шаг 2.1. Данные о расходе энергоресурсов за период с 2000 г., хранящиеся в специализированной информационной системе на базе *MySQL*, импортируются в табличный процессор *MS Excel*, а затем в пакет статистической обработки и анализа данных *STATISTICA*. Автоматизация данного шага обеспечивает постоянное обновление информации в модели, и, следовательно, возможность коррекции регрессионных зависимостей с учетом новых данных.

Шаг 2.2. Статистические данные обрабатываются средствами пакета *STATISTICA*, и полученные зависимости норм расхода энергоресурсов от выработки импортируются в модель. Автоматизация этапа статистической обработки данных обеспечивает снижение трудоемкости данного шага и позволяет пользоваться системой лицам, не являющимся специалистами в области моделирования и статистической обработки данных.

Этап 3. Решение оптимизационной задачи. Для оптимизации используется программное обеспечение *SOLVER* табличного процессора *MS Excel*. Оптимальное распределение производственной нагрузки по цехам ДМТ ЗОСа, обеспечивающее минимизацию себестоимости выпускаемой продукции, реализуется следующей последовательностью шагов.

Шаг 3.1. Вводятся зависимости расхода энергоресурсов от выработки, полученные для каждого цеха ДМТ ЗОСа на основе обработки статистических данных. Для основного сырья, вспомогательных материалов и постоянных затрат данные импортируются из системы "1С: Предприятие".

Шаг 3.2. Вводится суточный план выпуска ДМТ, который используется в качестве ограничения при расчете оптимальной загрузки цехов. В качестве ограничений при оптимизации используются также минимальная и максимальная суточная производительность каждого цеха.

Шаг 3.3. Вводится загрузка для каждого цеха, используемая в качестве начального приближения при оптимизации.

Шаг 3.4. Рассчитывается произведение расхода материалов и энергоресурсов на цену материала или энергоресурса. Вводится формула для расчета себестоимости ДМТ для каждого цеха как сумма затрат по статьям. В качестве целевой функции используется формула для расчета общей себестоимости по заводу.

Шаг 3.5. Минимизируется себестоимость продукции за счет изменения загрузки цехов, используя программное обеспечение *SOLVER* табличного процессора *MS Excel*.

На рис. 4 представлена структурная схема программного комплекса для моделирования производства ДМТ. Программный комплекс состоит из модулей импорта данных, статистической обработки данных, регрессионного анализа, модели распределения ресурсов и расчета загрузки цехов и себестоимости выпускаемой продукции.

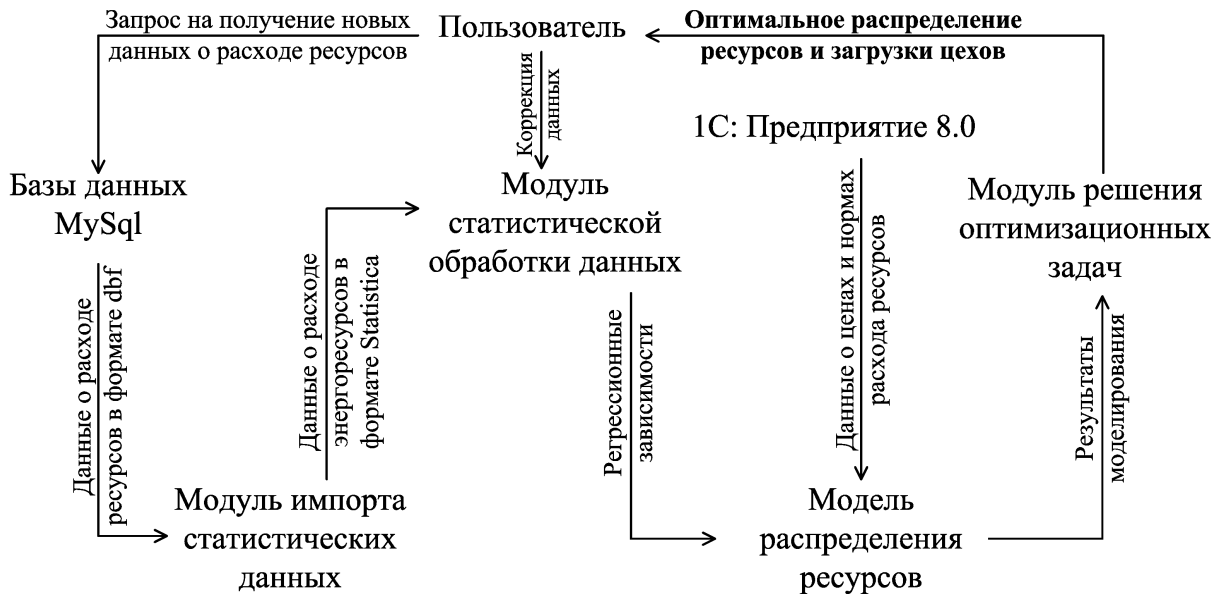


Рис. 4. Структурная схема программного комплекса

Для обеспечения работоспособности модели требуются текущие данные по ценам на материалы и ресурсы, нормам расхода материалов. Эти данные импортируются в модель из КИС "1С: Предприятие 8.0". Статистические данные по расходованию энергоресурсов импортируются из информационной системы предприятия на основе СУБД *MySql*. Данные по расходованию энергоресурсов до передачи в модель импортируются в модуль статистической обработки, в котором осуществляется предварительный анализ данных для исключения недостоверных значений и построения регрессионных зависимостей. Полученные регрессионные зависимости вводятся в модель и используются для оптимизации распределения загрузки цехов с целью снижения себестоимости выпускаемой продукции. Регрессионные зависимости, полученные на этапе эксплуатации модели, используются также в имитационной модели непрерывного производственного процесса ДМТ для исследования режимов производства ДМТ при режимах загрузки цехов, которые не исследованы в реальных условиях действующего производства.

Заключение

Обеспечение работоспособности модели в условиях промышленного предприятия требует обновления исходных данных и корректировки модели. Ввод и обработка данных из информационной системы промышленного предприятия вручную не представляется возможной из-за большого объема данных. Автоматизация импорта исходных данных в имитационную и регрессионную модели производственного процесса и пакет статистической обработки позволяют поддерживать модели производственного процесса в актуальном состоянии при изменении технологии производства.

THE SOFTWARE OF INTEGRATION IMITATING MODELS IN SOFTWARE PACKAGE BELSIM WITH COMPLEX INFORMATION SYSTEM

A.I. YAKIMAU, K.V. ZAHARCHANKAU

Abstract

The description of a software package *BelSim*, schemes of statement of imitating experiments is presented. The software *Data Integrator* is developed for integration of imitating model with complex information system of the enterprise, features of integration on a practical example of the decision of a problem of distribution of industrial loading are shown.

Литература

1. Якимов А.И., Альховик С.А. Имитационное моделирование в ERP-системах управления. Минск, 2005.
2. Якимов А.И. // Изв. Гомельского гос. ун-та им. Ф. Скорины. 2003. № 3 (18). С. 106–110.
3. Как программировать на XML / Х. М. Дейтел и др. Пер. с англ. М., 2001.
4. Альховик С.А., Якимов А.И. // Вестн. Могилевского гос. техн. ун-та. 2004. № 2 (7). С. 11–16.
5. Якимов А.И. // Изв. Гомельского гос. ун-та им. Ф. Скорины. 2005. № 5 (32). С. 90–92.
6. Якимов А.И., Захарченков К.В. // Компьютеризация в химической и смежных отраслях промышленности: Сборник материалов междунар. науч.-техн. семинара. Новополоцк: ПГУ, 2006. С. 47–51.
7. Якимов А.И., Захарченков К.В., Петров Р.В. // Изв. Гомельского гос. ун-та им. Ф. Скорины. 2005. №5 (32). С. 135–137.
8. Якимов А.И., Захарченков К.В. // Изв. Белор. инж. акад. 2005. №1(19)/2. С. 36–39.
9. Разработка динамической производственно-экономической модели завода органического синтеза ОАО "Могилевхимволокно". Отчет о НИР (заключ.) / Бел.-Рос. ун-т; рук. А.И. Якимов; исполн. С.А. Альховик и др.; № ГР 20066024. Инв. № ХД 0663. Могилев, 2007.