

денных изображений к общему количеству изображений, выданных пользователю в качестве результата. В данном случае важно максимизировать полноту при заданной точности.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Turk M., Pentland A. Eigenfaces for Recognition. *J. Cognitive Neuroscience*, 3(1):71-86, 1991.
2. Wang H., Li S.Z., Wang Y. Face Recognition under Varying Lighting Conditions Using Self Quotient Image. In FGR, 2004, pp. 819-824.
3. Zhou S. K., Zhao W., Tang X., Gong S. *Analysis and Modeling of Faces and Gestures*, Proc. of the Third International Workshop, AMFG 2007, Rio de Janeiro, Brazil, October 20, 2007, vol. 4778 of Lecture Notes in Computer Science. Springer, 2007.
4. Belhumeur P., Kriegman D. What is the Set of Images of an Object under All Possible Illumination Conditions, *IJCV*. 28(3), 1998, 245-260.
5. Georghiades A., Belhumeur P., Kriegman D. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643-660, 2001.
6. Spacek L. Collection of Facial Images. <http://cswww.essex.ac.uk/mv/allfaces/index.html>.
7. Martinez A., Benavente R. The AR Face Database. CVC Technical Report. 1998.

УДК 519.007

А.Н. Шабельников, В.И. Сачко, А.В. Евдокимов**РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПОДСИСТЕМ
ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В КОНТРОЛЬНО-
ДИАГНОСТИЧЕСКИХ КОМПЛЕКСАХ АВТОМАТИЗИРОВАННЫХ
СИСТЕМ УПРАВЛЕНИЯ***

На основе предложенного в работе [1] метаязыка структуризации данных авторами разработано программное обеспечение, позволяющее оптимизировать процесс обработки информации в контрольно-диагностическом комплексе (КДК) подсистем поддержки принятия решений автоматизированных систем управления (ПППР АСУ) на железнодорожном транспорте. Программная система использует модели систем и сетей массового обслуживания (СМО и СеМО, соответственно), адаптированные к новым информационным технологиям обработки информации в ПППР АСУ и имеет ряд преимуществ по сравнению с аналогами:

- ◆ наличие графического интерфейса, позволяющего пользователю – специалисту в предметной области создавать структуру СеМО и не требующего от пользователя сложных навыков программирования;
- ◆ программный комплекс разработан с учетом специфики предметной области и содержит набор функций, необходимых для моделирования работы подсистем, осуществляющих обработку информации в автоматизированных системах управления железнодорожными станциями;
- ◆ разработанный программный комплекс позволяет построить вычислительно-эффективную реализацию модели реальной системы;
- ◆ при разработке программного комплекса авторами была использована техника частичных вычислений [2], позволившая ускорить процесс компьютерных вычислений, что актуально для систем реального времени.

* Работа выполнена при поддержке РФФИ, грант № 03-07-90202.

Описание системы в терминах теории очередей. Протокольный сервер КДК ПППР АСУ осуществляет прием диагностической информации от всех подсистем и формирование в режиме реального времени баз данных событий, хранение и выдачу информации на АРМы обслуживающего персонала и другим абонентам. В системе предусмотрен прием сообщений о состоянии различных подсистем СС. Очевидно, что протокольный сервер представляет собой систему массового обслуживания. Классифицируем виды заявок, поступающих в систему:

1. Сообщение о состоянии оборудования. Поступает через примерно равные промежутки времени (но для различных видов оборудования промежутки времени могут отличаться). Результатом обработки данной заявки является сохранение информации о состоянии узла в базе данных. Обработка данного вида заявки не требует больших временных затрат.
2. Сообщение о состоянии системы (положении отцепа, и т. п.). Поступает через различные промежутки времени. Результатом обработки данной заявки является сохранение информации о состоянии узла в базе данных. Обработка данного вида заявки так же не требует больших временных затрат.
3. Запрос информации. Результатом обработки данной заявки является информация о состоянии системы или ее узла. Данные заявки поступают через случайные моменты времени. Обработка данного вида заявок требует более значительных временных затрат.

При моделировании системы будем исходить из следующих условий:

1. Одновременно система может обслуживать до 10 заявок. При поступлении одиннадцатой она отклоняется. Отклонение заявки является сбоем работы системы.
2. При одновременном обслуживании нескольких заявок время обслуживания увеличивается. Рассмотрим формулу, позволяющую оценить изменение времени обслуживания (рис. 1).

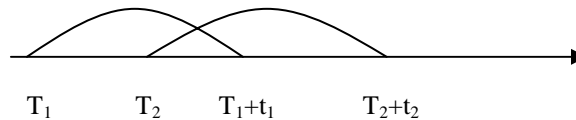


Рис. 1. Одновременное обслуживание заявок

Пусть T_1 – момент начала обслуживания заявки 1; T_2 – момент начала обслуживания заявки 2; t_1 – расчетная длительность обслуживания заявки 1 при условии, что она обслуживается одна; t_2 – расчетная длительность обслуживания заявки 2 при условии, что она обслуживается одна.

Тогда время одновременного пребывания двух заявок равно:

$$t_{12\text{одн}} = \min(T_1+t_1, T_2+t_2) - \max(T_1, T_2).$$

Расчетная длительность обслуживания первой заявки (с учетом одновременного обслуживания):

$$t_1' = t_1 + t_{12\text{одн}}/2 * r_1,$$

где r_1 – случайная величина в диапазоне от 1 до 2, характеризующая дополнительные затраты сервера, связанные с одновременным выполнением задач. Диапазон величины r_1 отражает следующие предположения:

а) две одновременно выполняющиеся заявки не могут выполняться быстрее, чем каждая из них в отдельности;

б) две параллельно выполняющиеся заявки выполняются не медленнее, чем две эти же заявки выполняющиеся последовательно.

$$t_2' = t_2 + t_{12\text{одн}}/2 * r_2,$$

где r_2 – случайная величина в диапазоне от 1 до 2, характеризующая дополнительные затраты сервера, связанные с одновременным выполнением задач.

Диапазон величины r_2 отражает следующие предположения:

а) две одновременно выполняющиеся заявки не могут выполняться быстрее, чем каждая из них в отдельности;

б) две параллельно выполняющиеся заявки выполняются не медленнее, чем две эти же заявки, выполняющиеся последовательно.

При создании описания данной системы узлами обслуживания заявки будут являться элементы сервера, такие как жесткий диск, сетевая карта, процессор. Следует отметить, что структура данной системы зависит от конкретной аппаратной и программной реализации сервера. Примером зависимости от аппаратной составляющей сервера может служить наличие одного или нескольких процессоров в системе, от чего зависит количество заявок обрабатываемых одновременно.

Примером зависимости структуры полученной системы от программной составляющей может служить использование на сервере различных систем управления базами данных (СУБД). Различные СУБД применяют отличающиеся друг от друга стратегии блокировки используемых ресурсов, что приводит к варьированию структуры получаемой модели. Этот факт использован авторами при разработке и внедрении методики оптимизации структуры базы данных протокольного сервера КСАУ СП.

Оптимизация структуры базы данных протокольного сервера КСАУ СП. Рассмотрим загрузку сервера баз данных (БД) при различных вариантах структуры БД (будем варьировать степень нормализации таблиц БД). При этом на данном этапе анализа, будем исходить из предположения, что объема оперативной памяти достаточно для хранения всех данных в таблицах и обращений к жесткому диску для чтения данных не требуется.

Рассмотрим два варианта таблиц и оценим время работы, затрачиваемое сервером для чтения и добавления данных при различных вариантах структуры БД.

Нормализованная структура представлена в виде табл. 1-3.

Таблица 1

Справочник узлов КСАУ СП – таблица SLNODES

Имя поля	Тип поля	Описание
CDNODE	Int, Primary key	Код узла
NMNODE	Varchar(255)	Наименование узла

Таблица 2

Справочник видов событий в системе КСАУ СП – таблица SLEVENT

Имя поля	Тип поля	Описание
CDEVENT	Int, Primary key	Код события
NMEVENT	Varchar(255)	Наименование события

Таблица 3

Таблица событий в системе КСАУ СП – таблица WORK

Имя поля	Тип поля	Описание
CD	Int, Primary key	
CDEVENT	Int, Foreign key (SLEVENT.CDEVENT)	Код события
CDNODE	Int, Foreign key (SLNODES.CDNODE)	Код узла
MOMENT	datetime	Момент наступления события
Pint_1	Int	Параметры события
Pint_2	Int	
Pint_3	Int	
Pflt_1	Float	
Pflt_2	Float	
Pflt_3	Float	

Плоская таблица БД будет иметь структуру, соответствующую табл. 4.

Таблица 4

Таблица событий в системе КСАУ СП – RAW_DATA

Имя поля	Тип поля	Описание
CD	Int, Primary key	
CDEVENT	Int	Код события
NMEVENT	Varchar(255)	Наименование события
CDNODE	Int	Код узла
NMNODE	Varchar(255)	Наименование узла
MOMENT	datetime	Момент наступления события
Pint_1	Int	Параметры события
Pint_2	Int	
Pint_3	Int	
Pflt_1	Float	
Pflt_2	Float	
Pflt_3	Float	

Результаты проведенного эксперимента приведены в табл. 5.

Таблица 5

Добавление данных – таблица результатов эксперимента

Количество строк	Время обработки			Среднее время
25000	6	8	6	6,67
50000	13	13	14	13,33
75000	20	20	19	19,33
100000	26	26	25	25,67
200000	51	51	52	51,33
300000	77	76	78	77
400000	108	107	107	107,33
500000	136	136	137	136,33

Результаты эксперимента дают численную оценку различных структур баз данных, что позволяет осуществить их оптимизацию.

Описание программного комплекса и его применения в системах поддержки принятия решений железнодорожного транспорта. В ходе исследования был разработан программный комплекс для имитационного моделирования КДК устройств сортировочной станции в виде СМО заданной конфигурации, использующий предложенную методику оптимизации структуры БД протокольного сервера.

Программный комплекс включает:

- ◆ модуль графического редактирования структуры СМО;
- ◆ модуль описания элементов (дизайнер объектов);
- ◆ модуль моделирования работы СМО;
- ◆ модуль вывода результатов моделирования.

Главное окно программы представляет собой экранную форму с меню (рис. 2). Меню содержит следующие пункты: «Файл», включающий подпункт «Выход», который позволяет выйти из программы; «Моделирование», включающий подпункт «Выполнить», который позволяет выполнить имитационное моделирование, полученной СМО; «Редактирование», включающий подпункт «Дизайнер объектов», позволяющий показать или скрыть окно дизайнера объектов.

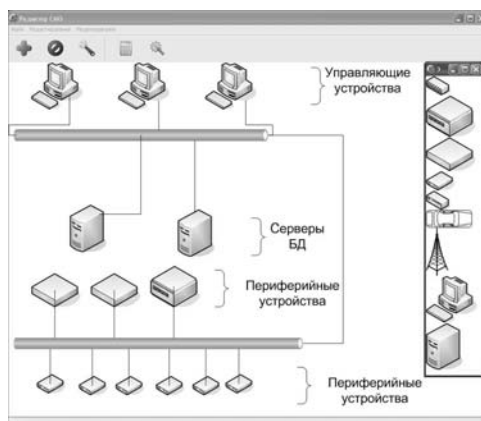


Рис. 2. Модель сервера КСАУ СП, построенная в программном комплексе

В окне редактирования возможны следующие действия:

1. Добавление элемента СМО.
2. Перемещение элемента СМО.
3. Удаление элемента.
4. Создание связей между элементами.

5. Редактирование свойств элемента. Для редактирования свойств необходимо, удерживая клавишу <Control>, нажать левой кнопкой мыши на выбранном элементе. При этом откроется окно дизайнера объектов.

Окно дизайнера объектов содержит две закладки, в которых находятся следующие поля редактирования основных свойств элемента.

«Максимальная длина очереди» позволяет изменить максимальную длину очереди элемента СМО.

«А=», «В=», «С=» – позволяет задать параметры формулы потока заявок.

«Формула» – позволяет задать формулу потока обслуживания заявок.

«Из файла» – позволяет выбрать файл, содержащий интервалы обслуживания заявок. Файл представляет собой последовательность чисел и содержит временные

интервалы обслуживания заявок. Каждый интервал времени начинается с новой строки.

Задание параметров осуществляется с использованием мастера настройки параметров, пример работы которого приведен на рис. 3.

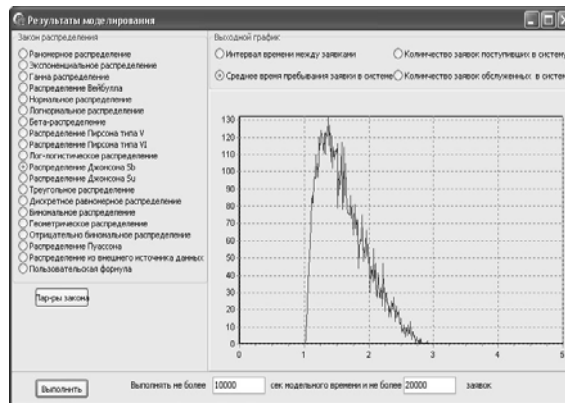


Рис. 3. Окно настроек параметров моделирования

После нажатия пункта «Выполнить» из меню «Симуляция» программа производит имитационное моделирование заданной системы и выдает окно «Результаты симуляции», в котором можно посмотреть численные параметры заданной СМО.

При двойном нажатии левой кнопкой мыши на выбранном элементе СМО возможно посмотреть графическую иллюстрацию работы заданного элемента СМО.

В данной версии выдаются следующие графики: «Длина очереди», «Средняя длина очереди», «Обработано заявок», «Среднее время пребывания в очереди», «Количество отказов», «Количество ошибочно обработанных заявок».

Разработанный программный комплекс позволяет повысить эффективность и качество обработки информации в действующих системах поддержки принятия решений автоматизированных систем управления сортировочным процессом, а также рассчитать необходимые характеристики устройств сбора и обработки информации в этих системах при их проектировании и модернизации.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Евдокимов А.В. Построение языка описания сложных объектов при проведении имитационного моделирования с использованием теории массового обслуживания // Обозрение прикладной и промышленной математики. – 2007, т. 14, вып. 6. – С. 1102.
2. Евдокимов А.В. Использование техники частичных вычислений для оптимизации процессов обработки информации в программных комплексах // Журнал научных публикаций аспирантов и докторантов. – Курск, № 2, февраль, 2008. – С. 184–188.