

УДК 681.518:658.512

Н.В. ТКАЧУК, С.В. ПОЛКОВНИКОВ, АЛЬ-ХАССАНИЕ ЗАХЕР

Национальный технический университет "ХПИ", Украина

МНОГОУРОВНЕВАЯ КОМПОЗИЦИОННАЯ СХЕМА МОДЕЛИРОВАНИЯ И ОЦЕНКИ КОМПОНЕНТНЫХ ПРОГРАММНЫХ РЕШЕНИЙ

Рассмотрен многоуровневый комплексный подход к моделированию и оценке функционирования компонентных программных решений (КПР) в составе информационно-управляющих систем (ИУС). Предложена иерархическая схема моделей, на каждом из логических уровней которой определены структура, нотация и механизмы моделирования, адекватные особенностям построения и технологиям реализации сложных КПР. Для практического применения данного подхода разработана архитектура и программно реализованы компоненты специального технологического инструментария.

информационно-управляющие системы, качество программного обеспечения, компонентное программное решение, имитационные модели, метрики, технологический инструментарий

Введение

На современном этапе развития индустрии информационных технологий решающее значение приобретают такие факторы, как *производительность* (performance) и *надежность* (reliability) разрабатываемого ПО [1 – 4]. Именно эти два показателя входят в список шести основных базовых характеристик *качества ПО* (software quality), которые перечислены в стандарте ISO / IEEE, а также опубликованы на официальном Web-сайте международного проекта SWEBOK (*Software Engineering Body of Knowledge*) www.swebok.org, представляющего собой информационный ресурс по вопросам программной инженерии, а именно:

- 1) *функциональность* (functionality),
- 2) *надежность* (reliability),
- 3) *удобство применения* (usability),
- 4) *производительность* (performance),
- 5) *сопровожаемость* (maintainability),
- 6) *переносимость* (portability).

Для определения количественных значений всех этих характеристик существует весьма большое число различных подходов и метрик, достаточно полный и интересный обзор которых можно найти, например, в работе Е.М. Лаврищевой и А.М. Рож-

нова [3]. При этом отмечается, что каждый из показателей (1) – (6), в свою очередь, определяется многими параметрами соответствующей программной системы и, кроме того, они являются взаимосвязанными величинами: такие, например, как производительность и надежность, удобство применения и функциональность и т.д.

Целью данной научной статьи является разработка нового комплексного подхода к моделированию и оценке КПР в процессах проектирования и сопровождения ИУС. Для достижения этой цели предлагается решить следующие *задачи*: 1) разработать концептуальную схему для комплексного моделирования особенностей построения и функционирования КПР; 2) программно реализовать соответствующий технологический инструментарий для практического применения предложенного подхода.

Актуальность проблемы исследования сложных КПР и некоторые существующие подходы к ее решению

Решение проблемы повышения эффективности и надежности функционирования ПО является достаточно сложной задачей, поскольку: 1) механизмы взаимодействия в таких распределенных программ-

ных архитектурах как *MS DCOM (Distributed Common Object Model)* и *CORBA (Common Object Request Broker Architecture)* [5] представляют собой слабо формализуемые задачи, при исследовании которых необходимо учитывать множество различных факторов и критериев; 2) в реальных условиях эксплуатации ИУС проведение каких-либо экспериментов затруднено, так как такие системы работают, как правило, в режиме реального времени, с непрерывным рабочим циклом, и их остановка либо перевод в режим тестирования чреваты опасностью возникновения нештатных ситуаций или значительными производственными потерями.

Поэтому для исследования и совершенствования разрабатываемых КПП различного назначения применяются подходы, связанные с моделированием альтернативных вариантов их реализации, изменением характера рабочей нагрузки и т.д. При этом используются различные типы моделей функционирования сложных распределенных систем: аналитические, метрические, имитационные модели (см., например, в [2, 4]). Весьма важным критерием эффективности таких подходов является адекватный выбор соответствующих информационных технологий для реализации таких моделей, поэтому наибольший эффект достигается тогда, когда для исследования тех или иных КПП создается *модельно-технологический комплекс* (МТК). Такой МТК должен обеспечивать решение задач моделирования и оценки КПП на всех этапах их жизненного цикла: начиная с проектирования и заканчивая отладкой и внедрением.

Концептуальная схема иерархии моделей для оценки характеристик КПП

Анализ особенностей функционирования КПП в составе сложных ИУС, в том числе в составе комплексов АСУ ТП на объектах газодобывающих предприятий [6], позволяет сделать вывод о том, что

для их моделирования и оценки необходима разработка комплексного подхода, который в контексте данной работы назван *многоуровневым композиционным подходом* (МКП).

Основная его идея заключается в том, чтобы рассматривать процесс моделирования производительности сложного КПП в виде многоуровневой схемы, в которой определены 4 типа абстракций моделирования, а именно: (I) – *доменное моделирование*: на этом этапе выполняется содержательный анализ предметной области (ПрО) разработки рассматриваемых КПП и определяются основные объекты моделирования, их характеристические свойства и характер связей между ними; (II) – *моделирование схем взаимодействия компонентов*: на этом уровне рассмотрения КПП определяются основные аппаратно-программные конфигурации, которые должны быть исследованы для оценки требуемых характеристик функционирования КПП, таких, как, например, их производительность и надежность; (III) – *моделирование функциональности отдельных программных компонентов*: на этом уровне детализации процесса анализа КПП разрабатываются модели функционирования их основных компонентов, причем как программных объектов (модели базовых сервисов, типовых приложений, модулей и т.п.) так и аппаратных составляющих КПП: модели каналов передачи данных, портов связи и т.д.; (IV) – *моделирование внутрикомпонентных процессов* в структуре отдельных слагаемых КПП: на этом уровне грануляции структуры исследуемого ПО должны быть проанализированы такие характеристики современных программных решений как многопоточные вычисления, различные механизмы управления ресурсами памяти (буферизация, кэширование), альтернативные схемы реализации транзакций и т.п.

Таким образом, в формализованном виде предлагаемая совокупность моделей $M^{(\Sigma)}$ определяется следующим кортежем:

$$M^{(\Sigma)} = \langle M^{(D)}, M^{(S)}, M^{(C)}, M^{(I)} \rangle, \quad (1)$$

где $M^{(D)}$ – множество моделей *доменного уровня*;

$M^{(S)}$ – множество моделей уровня схем меж-компонентного взаимодействия;

$M^{(C)}$ – множество моделей отдельных компонентов;

$M^{(I)}$ – множество моделей внутрикомпонентных механизмов.

На рис. 1 схематично показана концептуальная схема предлагаемого композиционного подхода для исследования производительности сложных КПП.



Рис. 1. Концептуальная схема МКП

Для каждого уровня абстракции такого рассмотрения КПП, соответственно, определяются три типа основных характеристик соответствующего модельного обеспечения:

1) *структура* (т.е. объекты и их отношения), необходимые для конструирования моделей данного уровня; 2) *формальная нотация*, вид которой должен позволять адекватно отражать статические и динамические характеристики соответствующих КПП; 3) *механизм моделирования* КПП на данном уровне: для этого помимо уже упомянутых выше математических, имитационных и метрических моделей, могут использоваться также информационно-логические, лингвистические, нечеткие модели и т.д., см., например, в [7]).

Более подробно механизмы моделирования на этих уровнях абстракции в общей схеме МКП могут быть определены следующим образом.

Уровень I. Доменная модель КПП: на этом уровне абстракции выполняется содержательный анализ ПрО, для которой разрабатывается КПП, и в виде результата должен быть получен структурированный объем знаний о ней в виде кортежа $M^{(D)}$, состоящего из двух множеств

$$M^{(D)} = \langle \Theta, P \rangle, \quad (2)$$

где Θ – доменная онтология [8] данной ПрО, причем каждая такая онтология может быть представлена, например, в виде аннотированной *диаграммы классов* в нотации UML [9];

$P = (p_1, p_2, \dots, p_k)$, $k \in Z^+$ – множество моделей *процессов обработки данных* в предметной области применения КПП; в качестве нотации для представления этих моделей может быть использована, в частности, структурная нотация IDEF3, и на рис. 2 показан пример одной из типичных диаграмм процессов с использованием этой нотации).

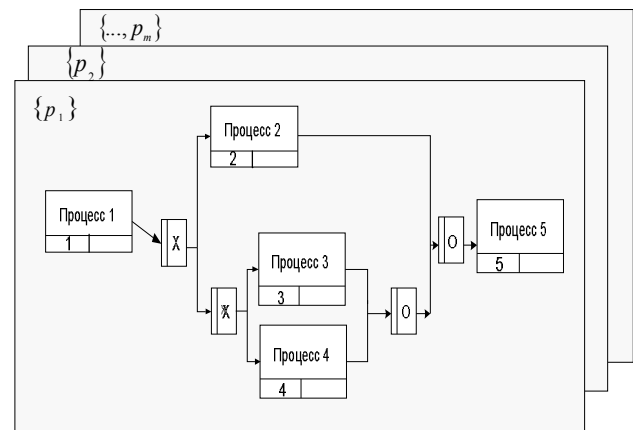


Рис. 2. Диаграммы процессов

Уровень II. Моделирование схем взаимодействия компонентов в составе соответствующего КПП: на этом уровне, на основе структурированной информации о доменной модели $M^{(D)}$ применения данного КПП (2), формируется кортеж $M^{(S)}$ в виде:

$$M^{(S)} = \langle C, \theta \rangle, \quad (3)$$

где $C = \{c_1, c_2, \dots, c_n\}, n \in Z^+$ – множество программных компонентов, входящих в состав соответствующих КНР;

$\theta = (\tau_1, \tau_2, \dots, \tau_n), n \in Z^+$ – множество структурных схем их взаимодействия, и при этом $\theta \subseteq P$.

Структурные схемы из множества θ могут быть, в свою очередь, представлены в виде диаграмм *кооперации объектов* (collaboration diagram) в нотации UML, и пример одной такой диаграммы показан на рис. 3.

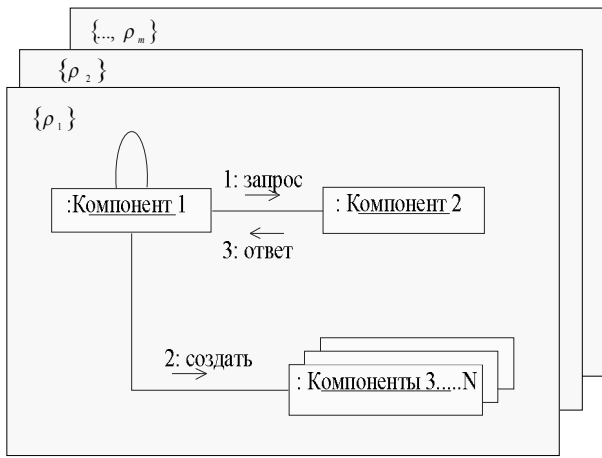


Рис. 3. Диаграммы действий

Именно этот тип диаграмм позволяет адекватно отразить событийно-ориентированный характер взаимодействия программных объектов в составе КНР.

В результате на этом уровне абстракции общей концептуальной схемы МКП могут быть получены модели, позволяющие делать выводы о влиянии его различных архитектурных особенностей на исследуемые характеристики КНР.

Уровень III. Моделирование функциональности отдельных программных компонентов: на этом уровне, на основе анализа информации об элементах множества C из выражения (3), для каждого из компонентов $c_i \in C$ определяется множество его функциональных действий:

$$\exists c_i \in C : \{f_1^{(i)}, f_2^{(i)}, \dots, f_m^{(i)}\} \quad m \in Z^+.$$

Адекватной нотацией для структурированного представления этой информации на данном уровне можно считать *диаграммы деятельности* (activity diagram) в формате языка UML, поскольку этот тип диаграмм позволяет эффективно отобразить такую весьма существенную особенность построения большинства реальных КНР, как *параллельные процессы*. Поэтому модельный механизм данного уровня для КНР может быть задан в виде кортежа $M^{(C)}$:

$$M^{(C)} = \langle C, F, \Lambda \rangle, \quad (4)$$

где C – множество программных компонентов, входящих в состав соответствующих КНР, см. выражение (3);

$$F = \bigcup_{i=1}^n \left(\bigcup_{j=1}^m f_j^{(i)} \right) - \text{множество их функциональных действий соответствующих компонентов;}$$

$\Lambda = \{\lambda_1, \lambda_2, \dots, \lambda_l\}, l \in Z^+$ – множество диаграмм действий, при этом $\Lambda \subseteq F \times F$; пример такой диаграммы показан на рис. 4.

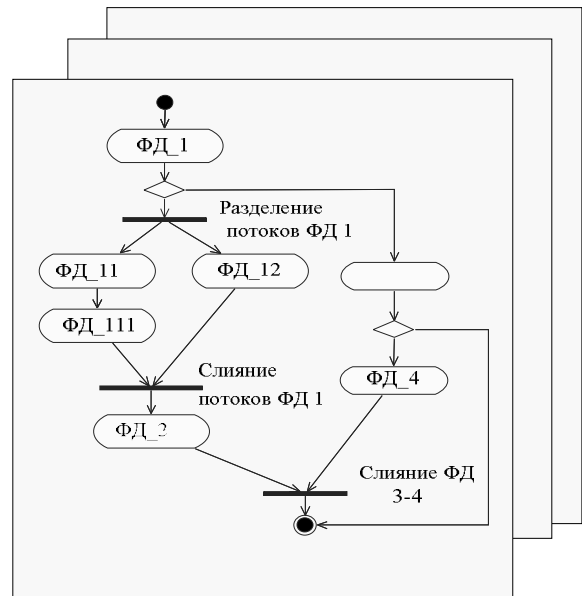


Рис. 4. Диаграммы действий для КНР

Уровень IV. Моделирование внутрикомпонентных программных механизмов – на нем каждый

программный компонент $\tau_i \in \theta, i \in Z^+$ рассматривается с учетом особенностей его внутренней программной реализации, например, с учетом возможных механизмов управления памятью, организации различных схем многопоточности, использования режима параллельных вычислений и некоторых других [10]. Целью моделирования на данном уровне абстракции является получение оценки метрик ресурсоемкости различных функциональных блоков, входящих в состав каждого из программных компонентов $\exists c_i \in C: \{b_1^{(i)}, b_2^{(i)}, \dots, b_k^{(i)}\}, k \in Z^+$. Этими функциональными блоками являются, например, потоки ввода / вывода данных, операции выделения / освобождения ресурсов памяти, участки программного кода, реализующие различные циклы, и т.п. Такой подход в литературе часто называют метрическим моделированием [2, 4], а в качестве графической нотации для представления структуры метрической модели КПП целесообразно использовать схемы различных графов выполнения программных компонентов. Они позволяют визуально представить пошаговое выполнение определенного участка программного кода и строятся на основе использования таких структурных примитивов, как: элементарный узел, цикл, узел выбора (с заданной вероятностью), расширяемый узел и некоторые другие [2]. Таким образом, модельный механизм данного уровня для КПП может быть задан также в виде следующего кортежа

$$M^{(I)} = \langle C, B, \Gamma \rangle, \quad (5)$$

где C – множество программных компонентов, входящих в состав соответствующих КПП, см. выражение (3);

$$B = \bigcup_{j=1}^n \left(\bigcup_{i=1}^k b_i^{(j)} \right) \text{ – множество их функциональных блоков;}$$

и

$\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_k\}, k \in Z^+$ – множество графов выполнения, пример такого графа показан на рис. 5.

Таким образом, приведенные концептуальные положения (I) – (IV) и формальные выражения (1) – (4) в своей совокупности представляют собой общую схему разработанного подхода к моделированию и оценке основных характеристик КПП.

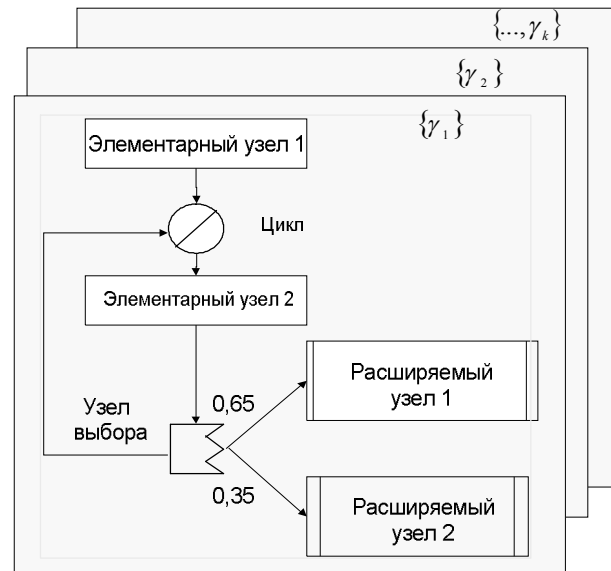


Рис. 5. Пример графа выполнения

Технологический инструментарий для реализации предложенного подхода

На рис. 6 показана архитектура разработанного технологического комплекса для реализации МКП.

Особенностью его реализации является то, что он представляет собой, по существу, интеллектуальную “надстройку”, которая создается для того, чтобы в проблемно-ориентированном режиме использовать такие хорошо известные и технологически эффективные средства моделирования сложных систем как, например, пакеты MATLAB, GPSS и некоторые другие [11]. Для этого в данном комплексе предусмотрены 3 основные группы функций:

1) задание внешних описаний структуры и алгоритмических особенностей КПП в виде набора idl-файлов (т.е. на языке описания программных интерфейсов, принятом в стандартах DCOM и CORBA [5]);

2) формирование на их основе модели КПП во внутреннем представлении;

3) конвертирование модели из внутреннего представления в формат, поддерживаемый другими моделирующими комплексами (MATLAB, GPSS и др.).

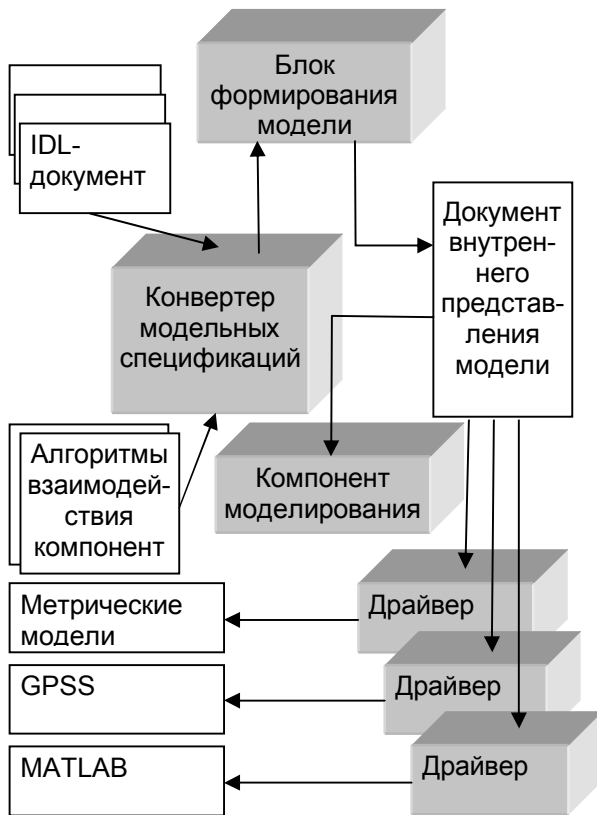


Рис. 6. Архитектура технологического комплекса

Для задания параметров соответствующих модельных описаний разработан диалоговый интерфейс пользователя, фрагмент программной реализации которого показан на рис. 7.

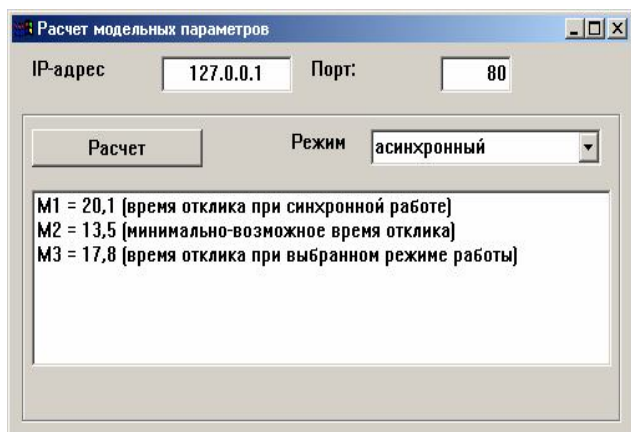


Рис. 7. Фрагмент диалогового интерфейса

С использованием элементов разработанного инструментария были проведены численные эксперименты для оценки производительности КПП, представляющего собой сервер обмена данными (СОД) в реальной SCADA-системе [12].

При этом исследовались факторы, которые оказывают влияние на величину времени отклика СОД (измеряемого в мс) и, в частности, влияние типа выбранной схемы многопоточности на быстродействие СОД при увеличении числа клиентских приложений в системе. Одна из полученных таким образом зависимостей показана на рис. 8.

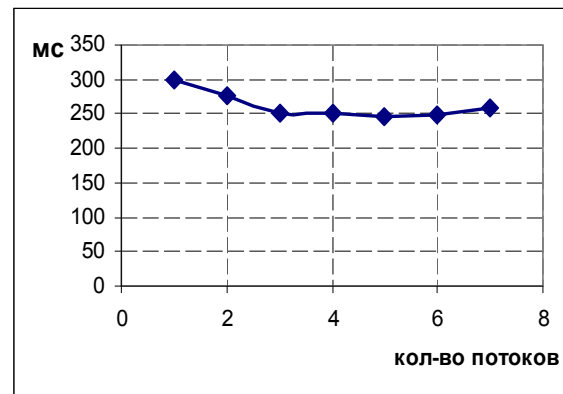


Рис. 8. График производительности работы КПП

Из нее можно сделать вывод о нелинейном характере такой зависимости и экспериментально определить наиболее эффективную конфигурацию данного КПП при условии необходимости его функционирования в заданных временных пределах (время отклика не более 300-400 мс).

Заключение

Представленный подход к моделированию и оценке производительности и надежности КПП позволяет рассматривать их особенности на различных уровнях детализации и при этом комплексно применять различные типы моделей их функционирования. Он может применяться как при разработке новых КПП, так и при анализе проблемных ситуаций и проведении процедур реин-

жиниринга унаследованных КПП в составе уже существующих ИУС [6, 12].

Продолжением этих исследований может быть постановка и решение комплекса задач, связанных с проблемой *адаптивного управления* работой сложных КПП в реальном масштабе времени [13]. В этом случае созданные механизмы моделирования процессов межкомпонентного взаимодействия могут быть использованы в контуре обратной связи управления таких комплексов, с целью выбора различных структурных схем КПП при изменении параметров операционной среды их функционирования.

Литература

1. Андон Ф.И., Коваль Г.И., Коротун Т.М. и др. Основы инженерии качества программных систем. – К.: Академперіодика, 2002. – 502 с.
2. Смит К., Уильямс Л. Эффективные решения: практическое руководство по созданию гибкого и масштабируемого программного обеспечения: Пер. с англ. – М.: Вильямс, 2003. – 448 с.
3. Лаврищева Е.М., Рожнов А.М. Концепция аналитической оценки характеристик качества программных компонентов // Проблемы программирования. – 2004. – № 2-3. – С. 180-187.
4. Менаске Д.А., Алмейда В.А.Ф. Производительность Web-служб. Анализ, оценка и планирование: [Метрики, модели, методы]: Пер. с англ. – С.-Пб.: ДиаСофт, 2003. – 466 с.
5. Дунаев С. Технологии Интернет-программирования. – С.-Пб.: БХВ-Петербург, 2003. – 480 с.
6. Ткачук Н.В., Горелый А.В., Земляной А.А. Комплекс имитационных моделей для исследования компонентных программных решений в ИУС АСУ ТП // Вісник НТУ «ХПІ». – 2004. – № 18. – С. 145-152.
7. Базы данных. Интеллектуальная обработка информации / В.В. Корнеев, А.Ф. Гареев и др. – М.: Нолидж, 2000. – 352 с.
8. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. – С.-Пб.: Питер, 2000. – 384 с.
9. Леоненков А.В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose. – М.: ИНТУИТ.РУ, 2006. – 318 с.
10. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – С.-Пб.: БХВ-Петербург, 2002. – 608 с.
11. Емельянов В.В., Ясиновский С.И. Введение в интеллектуальное имитационное моделирование сложных дискретных систем и процессов. – М.: АНВИК, 1998. – 226 с.
12. Ткачук Н.В. Архитектура программных сервисов синхронизации данных в распределенной многоуровневой WEB-SCADA системе // Радіоелектронні і комп'ютерні системи. – 2003. – № 2. – С. 75-81.
13. Ткачук Н.В., Аль-Хассани З. Адаптивные технологии разработки и сопровождения информационно-управляющих систем: современное состояние и перспективы развития // Вісник Національного технічного університету "ХПІ". – 2005. – № 19. – С. 125-144.

Поступила в редакцию 30.10.2006

Рецензент: д-р техн. наук, проф. М.Д. Годлевский, Национальный технический университет «ХПИ», Харьков.