

С. В. КОВАЛЬЧУК, В. Г. МАСЛОВ

## **ИНТЕЛЛЕКТУАЛЬНАЯ ПОДДЕРЖКА ПРОЦЕССА КОНСТРУИРОВАНИЯ КОМПОЗИТНЫХ ПРИЛОЖЕНИЙ В РАСПРЕДЕЛЕННЫХ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫХ СРЕДАХ**

Рассмотрены способы организации интеллектуальной поддержки пользователя в процессе конструирования композитных приложений на основе онтологического подхода. На их основе разработана система интеллектуальной поддержки пользователей при решении задач компьютерного моделирования в области квантовой химии.

*Ключевые слова:* интеллектуальная система, экспертные знания, онтология, логический вывод, композитное приложение, распределенная среда.

**Введение.** Современный этап развития парадигмы eScience характеризуется тенденцией к объединению географически удаленных друг от друга групп исследователей для решения комплексных научных задач, что требует консолидации распределенных вычислительных ресурсов и хранилищ данных. Консолидация позволяет создавать и использовать композитные приложения, которые включают в себя распределенные компоненты, разработанные и поддерживаемые различными группами специалистов и доступные как сервисы в сети Интернет. Поэтому процесс разработки композитных приложений осложняется не только технологическими особенностями стыковки отдельных компонентов (совместимостью форматов и структуры входных и выходных данных), но и методологическими аспектами интерпретации результатов их работы (совместимостью научных подходов, методов, моделей и алгоритмов). По-видимому, экстенсивным путем преодолеть проблему стыковки отдельных компонентов возможно лишь: а) в рамках конкретной задачи, для решения которой используется ограниченное число компонентов, ранее

известных пользователю; б) в ситуациях, когда все компоненты принадлежат одному и тому же разработчику, т.е. их технологическая и методологическая совместимость изначально обеспечена. В остальных случаях эффективное решение данной проблемы возможно только с использованием интеллектуальных технологий, основанных на отчуждаемых знаниях о природе и логической структуре программных компонентов композитного приложения [1].

Современные подходы к интеллектуальной поддержке процесса разработки композитных приложений на основе готовых компонентов (доступных в форме сервисов через Интернет) во многом наследуют особенности методологии Semantic Web [2]: отдельные компоненты рассматриваются в рамках единого пространства знаний в форме онтологии, что позволяет единообразно описывать правила их использования и сопряжения [3, 4]. Однако при таком подходе не уделяется должного внимания интеллектуальной поддержке пользователя в процессе построения самого композитного приложения. Оно может разрабатываться вручную (на основе аннотированных сервисов), с помощью шаблонов, а в ряде случаев — автоматизированно в ходе диалога с пользователем. Поскольку практическая цель построения композитного приложения определяется пользователем исходя из специфики предметной области, дальнейшее развитие технологий интеллектуальной поддержки возможно лишь в рамках симбиотического подхода, учитывающего особенности изучаемых процессов, специфику применяемых методов, алгоритмов и данных, а также отдельные аспекты их программной реализации и исполнения в распределенной вычислительной среде [5].

В настоящей работе подход к созданию системы интеллектуальной поддержки для разработки композитных приложений рассматривается на примере проблемной области, связанной с квантово-химическими расчетами и *ab initio* моделированием атомно-молекулярных наноразмерных структур и комплексов. Данный выбор обусловлен как практической значимостью этого направления в свете развития nanoиндустрии в целом, так и разнообразием входящих в него подходов, методов и технологий, обеспечивающих различные конкурирующие пути решения одной и той же задачи. В качестве среды для разработки и управления композитными приложениями используется многофункциональная инструментально-технологическая платформа CLAVIRE. Использование в ней проблемно-ориентированного языка описания композитных приложений EasyFlow [6] позволяет ограничить задачу интеллектуальной поддержки автоматизированным формированием соответствующего шаблона, на основе которого автоматически генерируется сценарий запуска в распределенной среде.

**Онтологический подход к описанию задач предметной области.** Первопринципное моделирование атомно-молекулярных наноразмерных структур и комплексов, в первую очередь, ориентировано на методы расчета оптических свойств, термодинамических характеристик, электрофизических и механических свойств материалов на их основе. Это приводит к необходимости использования различных методов описания свойств возбужденных состояний и электронных переходов, в том числе метода нестационарной теории функционала плотности (TDDFT), методов связанных кластеров применительно к возбужденным состояниям (EOM(LR)-CCSD), методов решения задач о нормальных колебаниях атомно-молекулярных систем методами квантовой химии, а также метода неравновесной функции Грина (NEGF). Учитывая механические свойства нанообъектов, необходимо также рассмотреть задачи оптимизации геометрии с ограничениями и обсудить особенности применения таких задач для моделирования поведения нанообъектов при механическом воздействии. Для систем очень большого размера целесообразно использовать полуэмпирические методы, а также методы QM/MM (квантовой механики/молекулярной механики), фрагментов и DFTB (функционала плотности сильной связи).

Обилие специфических задач и реализующих их методов приводит к необходимости выделить базовые понятия (*концепты*), позволяющие структурировать соответствующие знания. Исходя из общих принципов компьютерного моделирования можно выделить пять базовых концептов.

1. **Объект моделирования** — ключевой концепт компьютерного моделирования, нацеленного на определение некоторых характеристик данного объекта за счет использования одной или нескольких моделей, ассоциированных с ним. Объект моделирования описывается набором характеристик, которые условно можно разделить на две группы: *определяющие* — характеристики, фиксирующие объект моделирования и отличающие его от других объектов данного класса; *переменные* — характеристики, описывающие состояние объекта моделирования и чаще всего являющиеся входными или выходными данными для процесса моделирования.

2. **Модель** — служит для получения знаний об объекте моделирования. Представляет собой некоторую математическую или логическую структуру, отражающую заданный набор характеристик исследуемого объекта. Для работы с моделью могут быть применены различные методы моделирования, обеспечивающие, например, возможность получения по одним характеристикам модели других.

3. **Метод** — императивное описание процедуры работы с моделью для получения характеристик, проверки гипотезы и т.п. Именно методы чаще всего используются в качестве базового объекта для реализации в составе вычислительных пакетов и представляют наибольший интерес для пользователей этих пакетов. Очевидно, что может существовать несколько реализаций одного и того же метода.

4. **Реализация метода** — алгоритм, включенный в состав доступного пользователю вычислительного пакета и ассоциированный с некоторым методом.

5. **Вычислительный сервис** — вычислительный пакет, установленный на выделенном ресурсе в распределенной среде, используемый в режиме удаленного доступа.

На рис. 1 частично приведена структура онтологии, построенной на основе перечисленных выше концептов. Можно выделить два основных слоя: описание концептов (классов) и индивидов, реализующих концепты. Индивиды могут быть связаны отношениями, определенными на уровне концептов. Кроме того, допустимы отношения между отдельными концептами (например, отношение генерализации). В простейшем случае возможно ограничиться двумерными отношениями, что зачастую приводит к необоснованному усложнению структуры множества отношений за счет декомпозиции семантически связанных многомерных отношений на совокупность двумерных отношений.

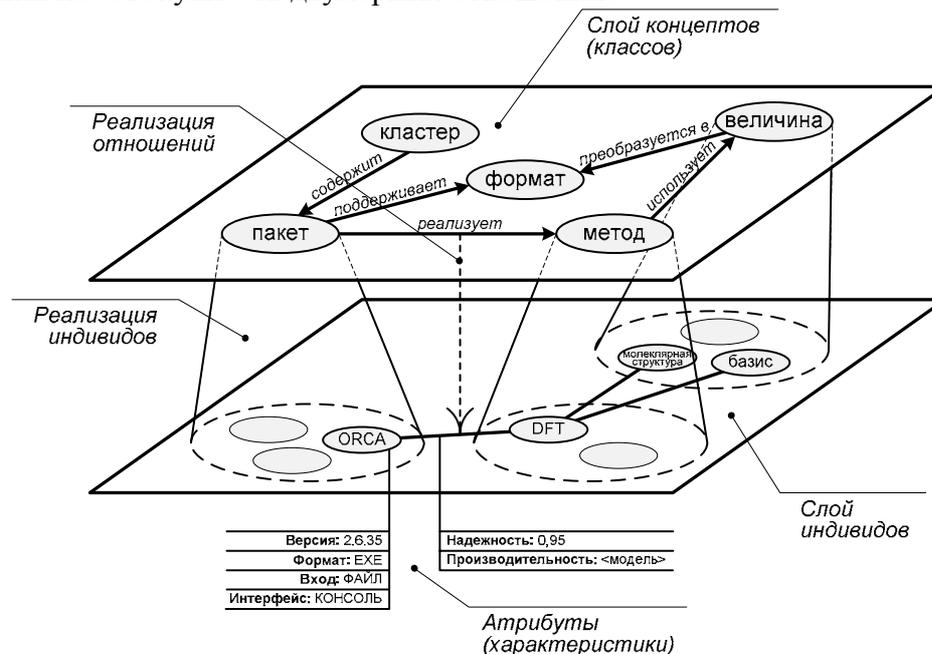


Рис. 1

Одним из элементов онтологии являются атрибуты (характеристики) индивидов, детализирующие их описание. Кроме того, одним из возможных расширений классического подхода

является ассоциация характеристик не только с индивидами (как реализациями классов), но и со связями между ними (как реализациями классов допустимых связей). Такое расширение применимо при введении операций второго порядка (например, анализе „неидеальности“ различного рода для связей, интерпретируемых как экспертные знания). В частности, из рис. 1 видно, что пакет ORCA [7] реализует метод функционала плотности (DFT), использующий в качестве параметров такие объекты, как собственно молекулярная структура и базис, по которому раскладывается решение. Характеристики связи между методом и пакетом позволяют оценить качество реализации (в данном случае в соответствии с критерием надежности), а также получить доступ к модели производительности, соответствующей данной реализации, для последующего использования в процессе моделирования.

Формально слой классов онтологии определяется как граф  $O = \langle C, R \rangle$ , где  $C$  — множество классов,  $R$  — множество абстрактных отношений, связывающих классы. Аналогично слой индивидов онтологии определяется как граф  $\tilde{O} = \langle \tilde{C}, \tilde{R} \rangle$ , где  $\tilde{C}$  — множество индивидов, а  $\tilde{R}$  — множество отношений между индивидами. Для каждого элемента слоя индивидов определены:

а) отношение генерализации  $gn^{(C)} : \tilde{C} \rightarrow C$ ,  $gn^{(R)} : \tilde{R} \rightarrow R$ , определяющее связь индивидов и связей между ними с соответствующими классами и связями классов;

б) „сторожевые“ условия (guard condition), определяющие применимость элементов в данных условиях

$$gc^{(C)}(F) : \tilde{C} \rightarrow \{0,1\}, \quad gc^{(R)}(F) : \tilde{R} \rightarrow \{0,1\},$$

где  $F$  — множество активных фактов, определенных для текущей задачи;

в) функция критериальной оценки

$$k^{(C)}(F) : \{ \tilde{c} \in \tilde{C} \mid gc^{(C)}(\tilde{c}) = 1 \} \rightarrow \Psi^{(C)}, \quad k^{(R)}(F) : \{ \tilde{r} \in \tilde{R} \mid gc^{(R)}(\tilde{r}) = 1 \} \rightarrow \Psi^{(R)},$$

где  $\Psi^{(C)}$  и  $\Psi^{(R)}$  — соответственно пространство критериев оценки индивидов и отношений между ними.

В общем случае сторожевые условия могут рассматриваться как один из критериев оценки. Здесь они выделены в отдельный класс в целях упрощения процедуры критериальной оценки, оперирующей сокращенными множествами  $\tilde{C}$  и  $\tilde{R}$ , для обеспечения логического вывода — построения шаблона композитного приложения на основе анализа описанной выше онтологии.

**Организация логического вывода.** Используя рассмотренную выше онтологию, можно получить структурированную информацию о параметрах, наиболее значимых для решения поставленной задачи, а также оценить доступные пользователю варианты решения поставленной задачи. При этом вариант решения определяется как кортеж  $S = (s_1, s_2 \dots s_N)$  фиксированной структуры,  $i$ -м элементом которого являются множества вида

$$s_i = \{ \tilde{c} \in \tilde{C} \mid gn^{(C)}(\tilde{c}) = c_i \},$$

где последовательность классов  $c_i \in C$  и требований к множествам  $s_i$  определяет общую структуру решения. Для оценки построенного решения по системе критериев применяется анализ графа

$$\tilde{O}' = \langle \tilde{C}', \tilde{R}' \rangle : \tilde{C}' = \bigcup_i s_i \cup \tilde{C}_S,$$

где  $\tilde{C}_S$  — присоединенная система классов

$$\tilde{C}_S = \left\{ \tilde{c}_S \mid \tilde{c}_S \notin \bigcup_i s_i, \exists \tilde{c}_1 \in \bigcup_i s_i : rch(\tilde{c}_S, \tilde{c}_1) \right\},$$

$rch(\tilde{c}_1, \tilde{c}_2)$  — отношение достижимости на графе  $\tilde{O}'$ . Оценка  $\tilde{O}'$  производится в пространстве критериев  $\Psi$ , определяемом пересечением множеств критериев, описывающих пространства  $\Psi^{(C)}$  и  $\Psi^{(R)}$ . С другой стороны, задача ранжирования способов запуска требует определения отображения  $\psi: \Psi \rightarrow R$ , позволяющего вычислить интегральный критерий качества оцениваемого решения.

Используя логический вывод на базе онтологической структуры, возможно построить описания композитных приложений с привлечением экспертных знаний различных уровней абстракции. Представим шаги логического вывода.

1. Производится *формализация требований* пользователя к решаемой задаче: требования формируют базовый набор активных фактов, записанный в терминологии используемой базы знаний. Факты, предоставляемые пользователем, можно разделить на две допускающие пересечение группы: *определяющие* — набор фактов, на основании которого формируется базовая цепочка решения (достаточный набор для однозначного определения решаемой задачи); *дополнительные* — набор требований и ограничений, более точно определяющих специфику решаемой задачи. Базовый набор активных фактов расширяется на основании правил логического вывода, доступных системе. Полученный таким образом набор активных фактов (или несколько альтернативных наборов) используется в последующих процедурах.

2. *Фильтрация*. На основании построенного набора активных фактов производится оценка элементов онтологии в пространстве критериев качества, по которой проводится первоначальная фильтрация онтологии, позволяющая отсеивать элементы, неприменимые для решения поставленной задачи. Отфильтрованная таким образом онтология представляет собой ориентированный граф, в котором каждой вершине или ребру сопоставлено значение в пространстве критериев качества. В простейшем случае этот шаг позволяет выбрать допустимые в данной ситуации элементы онтологической структуры.

3. *Определение доступных методов*. В составе построенного графа могут быть выделены связанные подграфы, каждый из которых состоит из элементов онтологии, определяющих один из доступных методов предметной области. При этом доступность метод определяется: а) реализацией этого метода хотя бы одним из сервисов, доступных в распределенной среде; б) соответствием метода сформированному набору активных фактов (определяется на основании процедуры фильтрации (см. шаг 2)).

4. *Композиция*. На основании полученного набора доступных методов должен быть построен абстрактный сценарий решения поставленной задачи. Структурная связь методов определяется: а) комбинацией типовых схем взаимодействия (последовательный запуск, цикл, параллельный перебор, ветвление и пр.); б) соответствием зависимостей по данным (входные данные для каждого из реализуемых методов должны быть либо предоставлены пользователем, либо получены в ходе выполнения предыдущих методов). Простейшим случаем можно назвать последовательный запуск сервисов, позволяющий в конечном итоге получить требуемую характеристику.

5. *Оценка*. Каждый из построенных вариантов композитного приложения должен быть подвергнут автоматическому анализу, результатом которого является интегральная оценка, позволяющая: а) провести ранжирование альтернативных вариантов решения для автоматического выбора лучшего из них (по совокупности критериев) или предоставления выбора пользователю; б) отсеять заведомо варианты решения, не удовлетворяющие заданным критериям качества. При этом входными данными для построения интегральной оценки служат оценки элементов онтологии, входящих в состав подграфов, описывающих отдельные методы, используемые в решении (см. шаг 3).

В общем случае на каждом этапе могут порождаться дополнительные варианты решения. Данная процедура проводится итеративно на каждом из концептуальных уровней базы

знаний. Таким образом, строится дерево решений, структура которого определяется базовой иерархией. На заключительном этапе производится оценка каждого из листьев полученного дерева, на основании которой можно либо провести автоматический выбор оптимального (с точки зрения качества) решения, либо предоставить пользователю осмысленные и обоснованные результаты оценки с предложением осуществить самостоятельный выбор (например, установив какой-либо из критериев качества как приоритетный).

На рис. 2 приведен пример формирования дерева принятия решений, позволяющего выбрать вариант решения квантово-химической задачи одной точки методом Хартри—Фока с помощью пакета GAMESS [8] в среде Грид Национальной нанотехнологической сети (конкурирующего с пакетами ORCA и MOLPRO).

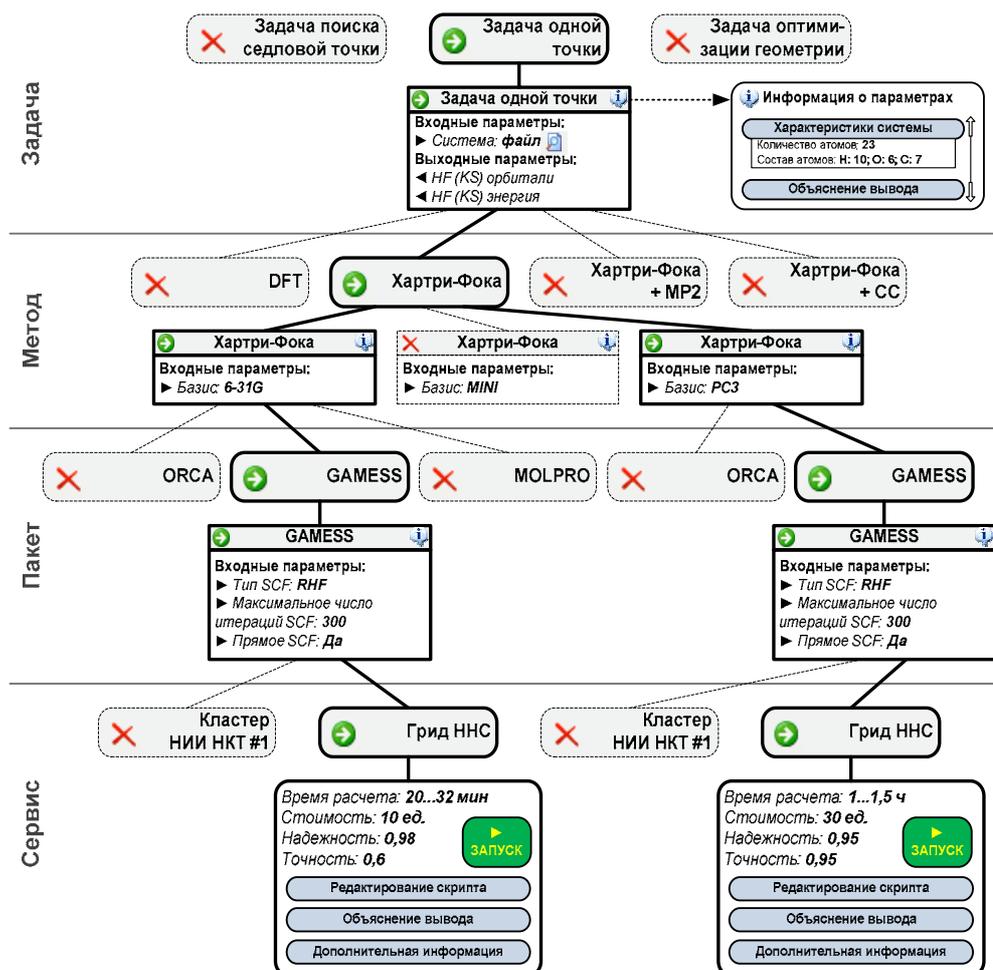


Рис. 2

На нижнем уровне рис. 2 приведены критерии качества формируемых решений.

1. *Точность предлагаемого решения*, определяемая с помощью экспертных знаний о: а) уровне точности используемых методов, б) качестве реализации этих методов в составе вычислительных пакетов.

2. *Время работы*, прогнозируемое на основании моделей производительности [9].

3. *Надежность решения*, основанная на истории запусков вычислительных сервисов, входящих в состав решения (в первую очередь, принимается во внимание доля отказов сервиса на конкретном ресурсе в распределенной среде).

Дополнительной характеристикой выбора может являться стоимость использования сервиса; на рис. 1 она определена в относительных единицах, которые могут иметь как временной, так и финансовый эквивалент.

**Программная реализация.** Описанный подход реализован в виде совокупности web-сервисов, которые могут использоваться для интеллектуальной и информационной поддержки специалистов в области квантовой химии и моделирования в нанотехнологиях, включая создание соответствующих композитных приложений, в частности, они обеспечивают:

- 1) логический вывод с целью поддержки принятия решения пользователем по выбору сервисов и формированию структуры композитных приложений;
- 2) информационно-справочную поддержку в области методов, задач и программного обеспечения в данной предметной области;
- 3) методическую поддержку, обучение специалистов рассматриваемой предметной области, осуществляемую в ходе анализа дерева принятия решений.

Программная реализация допускает два уровня хранения знаний — непосредственно в форме онтологической структуры и в форме реляционной базы данных. Онтологическая структура реализована с использованием возможностей языка OWL для описания базовых структур классов и индивидов и языка SWRL — для описания правил построения динамических связей между индивидами. База данных, хранящая фиксированный упрощенный вариант онтологической структуры и обеспечивающая более быстрый доступ к хранимым знаниям, функционирует на основе СУБД Microsoft SQL Server Compact Edition для хранения упрощенной онтологии, построенной на базе концептуальной иерархии индивидов (рис. 1). Использование базы данных позволяет за счет некоторого снижения гибкости обеспечить возможность ускорения взаимодействия с базой знаний. Автоматическое формирование базы знаний, синхронизированной по наполнению с текущей версией онтологической структуры, является одной из задач модуля обновления знаний.

Модуль логического вывода реализован в виде WCF-сервиса, предназначенного для построения дерева принятия решений на основе диалога с пользователем; для визуализации дерева принятия решений (например, в форме рис. 2) можно использовать различные инструментальные средства, в том числе в составе интерфейса прикладных комплексов и проблемно-ориентированных сред.

**Заключение.** Разработанный подход и программная реализация системы интеллектуальной поддержки позволяют строить и ранжировать различные варианты композитных приложений в ходе диалога с пользователем в различных предметных областях. Данная система применена в составе учебно-научного комплекса „Компьютерное моделирование в нанотехнологиях“ для поддержки разработки приложений комплекса НРС-NASIS II. Для этого использована база знаний в области квантово-химических расчетов и компьютерного моделирования в нанотехнологиях [10].

Работа выполнена в рамках проектов по реализации Постановлений № 218 и 220 Правительства Российской Федерации, при частичной поддержке государственного контракта № 16.647.12.2025 „Создание функционирующего в режиме удаленного доступа интерактивного учебно-методического комплекса для выполнения работ в области моделирования наноразмерных атомно-молекулярных структур, наноматериалов, процессов и устройств на их основе, в распределенной вычислительной среде“ и ФЦП „Научные и научно-педагогические кадры инновационной России на 2009—2013 гг.“.

#### СПИСОК ЛИТЕРАТУРЫ

1. The Fourth Paradigm. Data-Intensive Scientific Discovery / Ed. by T. Hey, S. Tansley, K. Tolle. Microsoft, 2009. 252 p.
2. Berners-Lee T., Hendler J., Lassila O. The Semantic Web // Scientific American Magazine. 2001. May. P. 29—37.
3. K-Wf Grid – Home [Электронный ресурс]: <<http://www.kwfgird.net/>>.

4. Chin G., Ruby Leung L., Schuchardt K., Gracio D. New Paradigms in Problem Solving Environments for Scientific Computing // Proc. of the 7th Intern. Conf. on Intelligent User Interfaces. 2002. P. 39—46.
5. Бухановский А. В., Васильев В. Н. Современные программные комплексы компьютерного моделирования e-Science // Изв. вузов. Приборостроение. 2010. Т. 53, № 3. С. 60—64.
6. Марьин С. В., Ларченко А. В., Ковальчук С. В., Князьков К. В., Болгова Е. В., Бухановский А. В. Интеллектуальные технологии распределенных вычислений для моделирования сложных систем // Науч.-техн. вестн. СПбГУ ИТМО. 2010. Вып. 70. С. 123—124.
7. Welcome to the ORCA [Электронный ресурс]: <<http://www.thch.uni-bonn.de/tc/orca/>>.
8. Gordon Group/GAMESS Homepage [Электронный ресурс]: <<http://www.msg.ameslab.gov/games/>>.
9. Марьин С. В., Ковальчук С. В. Сервисно-ориентированная платформа исполнения композитных приложений в распределенной среде // Изв. вузов. Приборостроение. 2011. Т. 54, № 10. С. 21—29.

#### Сведения об авторах

- Сергей Валерьевич Ковальчук** — канд. техн. наук; НИИ Научеомких компьютерных технологий Санкт-Петербургского государственного университета информационных технологий, механики и оптики; старший научный сотрудник;  
E-mail: kovalchuk@mail.ifmo.ru
- Владимир Григорьевич Маслов** — д-р физ.-мат. наук; Санкт-Петербургский государственный университет информационных технологий, механики и оптики, кафедра ЦИОТ;  
E-mail: maslov04@bk.ru

Рекомендована НИИ НКТ

Поступила в редакцию  
15.05.11 г.