

Общероссийский математический портал

Г. В. Рыбина, Ю. М. Блохин, Методы и программные средства интеллектуального планирования для построения интегрированных экспертных систем, *Искусственный интеллект и принятие решений*, 2018, выпуск 1, 12–28

Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочитали и согласны с пользовательским соглашением
<http://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 3.128.171.149

4 января 2025 г., 14:18:31



Методы и программные средства интеллектуального планирования для построения интегрированных экспертных систем¹

Аннотация. Рассматриваются вопросы разработки эффективных инструментальных средств для построения интеллектуальных систем, в частности, интегрированных экспертных систем (ИЭС). Анализируются общие проблемы создания современной инструментальной базы для поддержки разработки интеллектуальных систем, в контексте которых приводится описание базовой концепции интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ для поддержки построения ИЭС на основе задачно-ориентированной методологии. Исследуется постановка задачи интеллектуального планирования процессов прототипирования ИЭС. Описывается, основанный на эвристическом поиске, разработанный метод планирования действий инженеров по знаниям. Приводится пример использования интеллектуальной программной среды для разработки прототипа динамической ИЭС для управления медицинскими силами и средствами при крупных дорожно-транспортных происшествиях.

Ключевые слова: искусственный интеллект, интегрированные экспертные системы, задачно-ориентированная методология, модель интеллектуальной среды, инструментальный комплекс АТ-ТЕХНОЛОГИЯ, интеллектуальное планирование.

Введение

Среди отдельных классов прикладных интеллектуальных систем, применяющихся сегодня практически во всех производственных и социально значимых областях человеческой деятельности, одними из наиболее востребованных являются ИЭС, в которых в рамках единой *масштабируемой* архитектуры, совместно используется широкий спектр моделей и методов решения различных *неформализованных* и *формализованных* задач в статических и динамических проблемных областях. Анализ опыта разработки зарубежных и отечественных ИЭС, в том числе созданных на основе задачно-ориентированной методологии построения ИЭС (автор Г.В. Рыбина [1]) и поддерживающего эту методологию инструментального комплекса АТ-ТЕХНОЛОГИЯ, позволил выделить целый ряд общих научных и технологиче-

ских проблем, характерных для современных интеллектуальных систем в целом.

1. Существенное возрастание объемов информации в базах знаний (БЗ) и базах данных (БД) ИЭС приводит к значительному росту вычислительных затрат, увеличению трудоемкости привлечения экспертов к процессам получения, структурирования и верификации информации, в том числе извлеченной из других источников (ЕЯ-тексты, БД), и включающую, как правило, отдельные виды *недостовойной* информации (неопределенность, неточность, нечеткость и др.), а также *темпоральную* (временную) информацию.

2. Усложнение механизмов (средств) вывода решений в прикладных ИЭС, в частности, в динамических ИЭС связано со сложностью представления, хранения и анализа изменяющихся во времени данных, поступающих из внешних источников, а также выполнением темпоральных рассуждений о нескольких различных асинхрон-

¹Работа выполнена при финансовой поддержке РФФИ (проект 12-01-00467).

ных процессах (задачах) с учетом результатов моделирования внешнего мира и различных его состояний в реальном времени.

3. Значительное расширение типологии и масштабируемости архитектур прикладных ИЭС приводит к усложнению процессов *прототипирования* ИЭС за счет возрастания числа стадий и итераций в моделях жизненного цикла (ЖЦ) построения отдельных компонентов, необходимости интеграции средств вывода с разнородными средствами в составе ИЭС, которые в процессах вывода могут являться как источниками, так и потребителями данных и др.

4. Практика создания целого ряда статических, динамических и обучающих ИЭС на основе использования задачно-ориентированной методологии и поддерживающего эту методологию инструментального комплекса АТ-ТЕХНОЛОГИЯ [1-3], показала, что наибольшей сложностью по-прежнему обладают этапы *анализа системных требований* и *проектирования* ИЭС, причем существенное влияние оказывают специфика конкретной проблемной области (ПрО) и человеческий фактор, в частности:

- невозможность инженеров по знаниям (аналитиков знаний) полностью определять требования к разрабатываемым системам при использовании концепции «быстрого прототипирования» прикладных ИЭС;
- неприменимость традиционной технологии трассировки к БЗ ИЭС, невозможность использования методов модульной разработки и др.;
- отсутствие стандартного метода оценки качества верификации и валидации прототипов ИЭС;
- наличие достаточно высоких квалификационных требований к инженерам по знаниям, востребованность которых в ИТ-индустрии сегодня сравнима с программистами, однако проблемы их подготовки по-прежнему являются актуальными [4].

Соответственно, величина эффекта от создания и применения интеллектуальных систем различной архитектурной типологии напрямую зависит от использования высокотехнологичных *инструментальных средств* (ИС) и массовости их применения в стратегически значимых областях, что определяет особую актуальность и важность исследований и разработок, связанных с созданием мощной современной инструментальной базы для поддержки разработки интеллектуальных систем.

В этой связи следует отметить, что подавляющее число коммерческих зарубежных ИС, например G2 (Gensym Corp.) [26], SHINE (NASA / JPL) [27], RTXPS (Environmental Software & Services GmbH) [28] и др., а также наиболее успешные отечественные разработки последних лет (например, исследовательская облачная платформа IACPaaS [5, 6] и др.) не в полной мере удовлетворяют вышеизложенным требованиям. Это объясняется тем, что инструментальные средства не «знают» что именно проектирует и разрабатывает с их помощью инженер по знаниям, поэтому эффективность применения ИС полностью определяется искусством разработчиков. Кроме того, в настоящее время существует большое количество универсальных сред разработки, активно используемых в ИТ-индустрии (Microsoft Visual Studio, Eclipse, NetBeans, Apple XCode и др.), но не ориентированных на специфику создания интеллектуальных систем, поэтому их адаптация для автоматизации процессов разработки ИЭС является нецелесообразной и сопряжена со значительными трудностями.

В контексте решения вышеперечисленных проблем определенные результаты были получены в рамках KBSA-подхода (Knowledge Based Systems Assistant), где особое внимание уделялось выбору типов спецификаций, генерации кода из спецификаций и др. (например, PiES WorkBench [7] и др.), что обеспечивало некоторое *ассистирование* действий инженеров по знаниям в процессах проектирования. Однако наибольшие результаты в создании ИС нового поколения были получены в рамках KBSE-подхода (Knowledge Base Software Engineering), интегрирующего как минимум возможности средств поддержки разработки традиционных экспертных систем (ЭС) и CASE-средств, а как максимум, обеспечивающих *интеллектуализацию* процессов разработки интеллектуальных систем на основе знаний.

Одним из наиболее известных ИС нового поколения, в котором реализован KBSE-подход, в настоящее время является интеллектуальная программная среда комплекса АТ-ТЕХНОЛОГИЯ, обеспечивающая автоматизированную поддержку процессов построения ИЭС на основе задачно-ориентированной методологии [1], что включает, в том числе, интеллектуальное ассистирование (планирование) действий инженеров по знаниям. Детальное

описание модели интеллектуальной программной среды и ее компонентов содержится в [1], а здесь отметим только функциональные возможности, наиболее важные в контексте данной работы:

- автоматизация процессов прототипирования прикладных ИЭС различной архитектурной типологии (статических, динамических, обучающих) на всех этапах ЖЦ построения ИЭС;
- построение моделей архитектур проектируемых ИЭС на нескольких уровнях интеграции ЭС с другими компонентами ИЭС (БД, обучающие системы, имитационные модели и др.) с обеспечением внутренней интегрируемости по представлению, данным и управлению;
- накопление экспертного и технологического опыта инженеров по знаниям в специальных технологических БЗ в виде типовых проектных процедур (ТПП) и повторно-используемых компонентов (ПИК) предшествующих проектов с целью сокращения сроков проектирования и снижения интеллектуальной нагрузки на инженеров по знаниям;
- наличие удобного когнитивно-графического интерфейса построения всех компонентов ИЭС с учетом человеческого фактора, открытость, кроссплатформенность и переносимость в условиях клиент-серверной архитектуры.

Подобный подход к реализации инструментария является наиболее актуальным и хорошо соответствует современным тенденциям, связанными со снижением трудозатрат на разработку и смягчением квалификационных требований к проектировщикам и разработчикам прикладных систем. Причем существенных результатов в развитии интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ можно достичь за счет усиления роли ПИК в процессах анализа и проектирования, а также реинжиниринга структуры ТПП. Наиболее значительное повышение степени автоматизации связано с расширением функциональности базового компонента интеллектуальной программной среды - интеллектуального планировщика, что обеспечивает не только эффективную помощь инженерам по знаниям на отдельных этапах ЖЦ построения ИЭС различной архитектурной типологии, но и позволяет осуществлять планирование и управление разработкой в целом.

Следует отметить, что реализация текущей версии интеллектуального планировщика пред-

ставляет собой гибридизацию подходов, базирующихся на использовании HTN-формализма [23] и гибких механизмов поиска решений, применяемых в ИЭС, что позволяет использовать декларативный способ описания знаний о ходе разработки, в данном случае на языке представления знаний продукционного типа [1]. Однако с усложнением архитектур ИЭС, а также появлением в составе технологической БЗ большого количества ТПП и ПИК время поиска решений значительно увеличилось, поиск стал достаточно трудоемким, и негативный эффект от неоптимальности выбора решений стал более значительным. Поэтому возникла необходимость усовершенствования методов и алгоритмов планирования, используемых интеллектуальным планировщиком. Результаты комплексного анализа современных методов интеллектуального планирования [9-11] и проведенные экспериментальные исследования показали, в частности, целесообразность использования для решения данной проблемы достаточно известного подхода, связанного с *планированием в пространстве состояний* [12].

Таким образом, наиболее важными требованиями к методологиям и современным инструментариям для построения интеллектуальных систем, в частности ИЭС, обладающих мощной функциональностью и масштабируемой архитектурой, становятся факторы снижения стоимости-временных показателей разработки и интеллектуальной нагрузки на инженеров по знаниям за счет усиления степени интеллектуализации процессов построения систем на всех этапах ЖЦ, в том числе привлекая методы и средства интеллектуального планирования.

Целью данной работы является рассмотрение наиболее важных из полученных теоретических и технологических результатов, связанных с дальнейшим развитием и усовершенствованием базовых компонентов интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ.

1. Интеллектуальное планирование процессов прототипирования ИЭС на основе средств комплекса АТ-ТЕХНОЛОГИЯ

В работе [2] был проведен детальный систематизированный обзор современных исследований и разработок в области интеллектуального планирования, поэтому здесь ограничимся

только краткими сведениями, необходимыми для понимания общей постановки задачи в контексте рассматриваемых научно-технологических результатов.

Под *интеллектуальным планированием* чаще всего подразумевается автоматическая генерация планов некоторой программно-аппаратной системой, тем не менее сам термин не имеет однозначного определения. В русскоязычной литературе наиболее употребляемым является термин «интеллектуальное планирование» [9], а в англоязычной – «автоматическое планирование» (automated planning) [10], однако в обоих случаях подразумевается собственно процесс генерации плана вычислительной машиной. Соответственно, в контексте интеллектуального планирования *план* - это некоторое представление будущего поведения. В частности, план обычно представляет собой набор действий с некоторыми ограничениями (например, темпоральными) для исполнения некоторым агентом или агентами [10].

В области интеллектуального планирования к настоящему времени разработано значительное число методов, подходов, формализмов и т.д., среди которых следует выделить: планирование с помощью пропозициональной логики; планирование в пространстве планов; планирование в пространстве состояний; планирование как задача удовлетворения ограничений; планирование на основе прецедентов; трансляция в другую проблему; темпоральное планирование; планирование в недетерминированных и вероятностных областях; иерархическое планирование (HTN-формализм) и др. Детальные обзоры можно найти в работах [2, 9, 10, 13].

Достаточно разнообразным является и широкий спектр практического применения методов интеллектуального планирования для решения прикладных задач. Используя [2], выделим лишь наиболее востребованные области: управление автономными роботами [11] и беспилотными летательными аппаратами [14]; композиции веб-сервисов [15]; автоматизированное обучение (в частности, для построения индивидуальных планов обучения [16]); калибровка оборудования [17]; ресурсно-календарное планирование [18, 19]; оптимальное использование ресурсов во времени [20] и др. Тем не менее, следует отметить, что в целом использование методов и средств интеллектуального планирования с целью автоматизированной

поддержки процессов построения интеллектуальных систем является пока слабо исследованной областью.

В соответствии с [1] рассмотрим наиболее общую формальную постановку задачи планирования. Пусть $I = \{I_i \mid i \in N\}$ - множество всех описаний состояний рассматриваемой системы (подмножество окружающего мира). Тогда задача планирования представляется в следующем виде: $PlanTask = \langle I, E, G, A \rangle$, где $E \subset I$ - полное описание начального состояния; $G \subset I$ - описание (возможно, неполное) целевого состояния; $A = \{a_i \mid i \in N\}$ - множество действий (операторов, задач), изменяющих состояние системы. Упорядоченное множество $a_1, \dots, a_n, n \in N$ называется решением задачи планирования, если последовательность операций $a_n(a_{n-1}(\dots a_1(E)))$ приводит в итоге к целевому состоянию G . Данное упорядоченное подмножество $a_1 \dots a_n$ также называется планом.

Теперь конкретизируем общую постановку задачи планирования как формальную постановку задачи интеллектуального планирования применительно к моделям, методам и средствам, выступающим в качестве концептуальной основы задачно-ориентированной методологии построения ИЭС и интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ. Как было отмечено выше, основной целью данной методологии и поддерживающего ее инструментария является *интеллектуализация* достаточно сложных и трудоемких процессов *прототипирования* прикладных ИЭС различной архитектурной типологии на всех этапах ЖЦ, начиная от системного анализа ПрО до создания серии прототипов ИЭС. Для снижения интеллектуальной нагрузки на инженеров по знаниям, минимизации возможных ошибочных действий и временных рисков при прототипировании ИЭС предусмотрено использование специальной *технологической БЗ*, содержащей значительное число ТПП и ПИК, отражающих экспертный опыт инженеров по знаниям по созданию прикладных ИЭС (статических, динамических, обучающих).

Таким образом, исходя из требований задачно-ориентированной методологии и интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ, формальная постановка задачи интеллектуального планирования процессов прототипирования ИЭС может рассматриваться в контексте *модели процессов прототипирова-*

ния ИЭС в следующем виде: $M_{proto} = \langle T, S, Pr, Val, A_{IES}, PlanTask_{IES} \rangle$, где T - множество ПрО, для которых создаются прикладные ИЭС; S - множество стратегий прототипирования; Pr - множество созданных и/или создаваемых прототипов ИЭС на основе задачно-ориентированной методологии; $Val: T \times S \rightarrow \{0, 1\}$ - функция экспертной валидации прототипа ИЭС, определяющая необходимость и/или возможность создания последующих прототипов ИЭС для конкретной ПрО; A_{IES} - множество всех возможных действий инженеров по знаниям в процессе прототипирования; $PlanTask_{IES}: T \times S \times Pr \rightarrow P(A)$ - функция планирования действий инженеров по знаниями для получения текущего прототипа ИЭС для конкретной ПрО.

Следует отметить, что реализация таких компонентов M_{proto} как Pr , A_{IES} и $PlanTask_{IES}$ в наибольшей степени определяется используемой концепцией интеллектуальной программной среды, модель которой в соответствии с [1] представляется в виде $M_{AT} = \langle KB, K, P, TI \rangle$, где KB - технологическая БЗ о составе проекта и типовых проектных решениях, используемых при разработке ИЭС; $K = \{K_i \mid i \in N\}$ - множество текущих контекстов, состоящих из множества объектов из KB , редактируемых или выполняющихся на текущем шаге управления; P - специальная программа - интеллектуальный планировщик, управляющая процессами разработки и тестирования ИЭС; $TI = \{TI_i \mid i \in N\}$ - множество инструментов TI_i , применяющихся на различных этапах разработки ИЭС.

С точки зрения интеллектуальной поддержки процессов разработки ИЭС важное место отводится компоненту KB модели M_{AT} , выступающему в качестве информационного хранилища (технологической БЗ) в данной среде и определяемого как $KB = \langle WKB, SKB, PKB \rangle$, где WKB - БЗ, содержащая знания о ТПП, описывающих последовательности и способы применения тех или иных средств при создании прикладных ИЭС, а также последовательности этапов создания ИЭС; SKB - БЗ, включающая знания об использовании ТПП и ПИК, ранее созданных прототипов ИЭС; PKB (опционально) - БЗ, содержащая специфические знания, используемые на различных этапах создания прототипа ИЭС для решения задач, требующих нестандартного подхода.

Раскрывая более детально содержимое технологической БЗ, необходимо отметить, что

модель любой ТПП представляется в виде $TPP = \langle C, L, T \rangle$, где C - множество условий, при выполнении которых возможна реализация ТПП; L - сценарий выполнения, описанный на внутреннем языке описания действий ТПП; T - множество параметров, инициализируемых интеллектуальным планировщиком при включении ТПП в план разработки прототипа ИЭС.

Каждый ПИК, участвующий в разработке прототипа ИЭС, представляется как $PIK = \langle N, Arg, F, PINT, FN \rangle$, где N - имя компонента, под которым он зарегистрирован в комплексе АТ-ТЕХНОЛОГИЯ; $Arg = \{Arg_i \mid i \in N\}$ - множество аргументов, содержащих поддеревья БД текущего проекта, служащие входными параметрами для выполнения функций из множества $F = \{F_i \mid i \in N\}$ - множество методов (интерфейсов ПИК) данного компонента на уровне реализации; $PINT$ - множество наименований интерфейсов других ПИК, используемых методами данного ПИК; $FN = \{FN_i \mid i \in N\}$ - множество наименований функций, выполняемых данным ПИК. Таким образом, множество ПИК в совокупности с возможными сценариями их использования формируют компонент A_{IES} модели M_{proto} .

Основным процедурным (операционным) компонентом модели M_{AT} является интеллектуальный планировщик, который в общем виде описывается моделью $P = \langle SK, AF, Pa, Pb, I, GP \rangle$, где SK - состояние текущего контекста, при котором активизируется планировщик; $AF = \{AF_i \mid i \in N\}$ - множество функциональных модулей AF_i , входящих в состав планировщика; Pa - процедура выбора текущей цели на основании плана разработки; Pb - процедура выбора наилучшего функционального модуля-исполнителя из списка возможных кандидатов; I - процедуры, обеспечивающие интерфейс с соответствующими компонентами комплекса АТ-ТЕХНОЛОГИЯ; GP - процедуры работы с глобальным планом разработки ИЭС.

Теперь более детально рассмотрим компонент Pr модели M_{proto} , включающий множество прототипов ИЭС, модель архитектуры которых строится в соответствии с требованиями задачно-ориентированной методологии. Здесь на верхнем уровне интеграции компонентов проектируемых ИЭС, исходя из идей *глубинной интеграции* компонентов в модели архитектуры прототипа ИЭС [1], используется подход, связанный с расширением функциональности традиционных ЭС путем включения нетради-

ционных для них функций, реализуемых некоторым компонентом K (где K – это СУБД, ППП и т.д.). Поэтому средний уровень интеграции в модели архитектуры каждой конкретной ИЭС определяется спецификациями наборов функций, как простой ЭС (неформализованные операции), так и функций, не свойственных ЭС (формализованные операции), что в целом отражает состав и структуру всех компонентов ИЭС и их информационные и управляющие связи, а также место и роль неформализованных задач (НФ-задач) среди других задач. Все спецификации определяются на этапе анализа системных требований пользователя на разработку ИЭС и этапе получения знаний из различных источников знаний (при наличии НФ-задач).

Другой важный аспект построения модели архитектуры проектируемой ИЭС [1] связан с тем, что для отображения взаимосвязей реальной системы используются методологические средства структурного анализа, в результате чего модель архитектуры конкретизируется до уровня *расширенной информационно-логической модели* ($M_{РИЛ}$), представляющей собой совокупность диаграмм информационных потоков и описаний их элементов в графической нотации Гейна-Сарсона, которая дополнена специальным элементом - *неформализованная операция* (НФ-операция), указывающим на наличие НФ-задачи, что показывает на необходимость привлечения к процессам получения знаний конкретных экспертов или источников знаний других типов (БД, ЕЯ-тексты).

Поэтому основой модели архитектуры прототипа ИЭС ($M_{ИЭС}$) является расширенная информационно-логическая модель ($M_{РИЛ}$), а модель каждого прототипа ИЭС соответственно представляется семеркой вида [1]: $PRJ = \langle PN, KB_{PRJ}, Solver, PD, M_{РИЛ}, PPIK, PCOMP \rangle$, где PN – имя проекта, KB_{PRJ} – БЗ прототипа ИЭС, $Solver$ – машина (средства) вывода прототипа ИЭС; PD – данные проекта, т.е. информация различного характера (знания, данные, отдельные параметры, тексты и т.д.), используемая интеллектуальным планировщиком как в процессе разработки прототипа ИЭС, так и для генерации готового прототипа; $M_{ИЭС}$ – модель архитектуры прототипа на основе расширенной информационно-логической модели ($M_{РИЛ}$) в виде иерархии расширенных диаграмм потоков данных (РДПД); $PPIK$ – совокупность ПИК;

$PCOMP$ – совокупность различных подсистем ИЭС, разработанных как средствами комплекса АТ-ТЕХНОЛОГИЯ, так и с помощью внешних приложений.

Множество стратегий прототипирования (компонент S модели M_{proto}), как правило, включает в себя три базовые стратегии, детально описанные в [1] и других работах, а функция экспертной валидации прототипа ИЭС (компонент Val модели M_{proto}) выполняется непосредственно экспертом.

Как было отмечено выше, для эффективной реализации компонента $PlanTask_{IES}$ модели M_{proto} необходимо создание новой версии планировщика, используя планирование в пространстве состояний. Экспериментальное программное исследование показало, что наилучшие результаты достигаются, если пространство поиска формируется за счет моделирования действий инженера по знаниям при построении фрагментов модели архитектуры прототипа ИЭС с помощью соответствующих ТПП. Для формального описания этого процесса можно использовать теорию графов путем сведения к задаче *покрытия* модели архитектуры $M_{РИЛ}$, представленной в виде помеченного графа $G_{РИЛ}$, фрагментами ТПП в виде соответствующих подграфов.

При этом для реализации $PlanTask_{IES}$ важную роль играет план построения конкретного покрытия, т.е. последовательность примененных фрагментов ТПП, который, в свою очередь, может быть однозначно преобразован в план построения прототипа ИЭС. В этом случае план построения прототипа ИЭС можно представить в виде $Plan = \langle A_G, A_{atom}, R_{prec}, R_{detail}, PR \rangle$, где A_G – множество *глобальных* задач (декомпозируемых на подзадачи); $A_{atom} = \{atom_i \mid i \in N\}$ – множество *плановых* (атомарных) задач, выполнение которых необходимо для разработки прототипа ИЭС; $R_{prec}: A_{atom} \times A_{atom} \rightarrow \{0,1\}$ – функция, определяющая отношение предшествования между плановыми задачами; $R_{detail}: A_{atom} \rightarrow A_G$ – отношение, показывающее принадлежность плановой задачи к глобальной; PR – представление плана, необходимое для удобства восприятия плана инженером по знаниям.

Таким образом, с помощью отношения R_{prec} и множеств A_G и A_{atom} можно сформировать две сети задач - *укрупненную* и *детальную*. Укрупненная сеть задач, полученная с помощью R_{prec}

и A_G , называется *глобальным планом* (отношение предшествования между элементом A_G получено на основе отношения детализации R_{detail}), а детальная сеть задач, полученная на основе R_{prec} и A_{atom} , называется *детальным планом*. При этом каждая плановая задача связана с определенной функцией конкретного операционного ПИК. Следует отметить, что представление плана в простейшем случае является линейно упорядоченной последовательностью задач, а в более сложном - иерархией задач (по подзадачам, по этапам, по задачам, что является близким к HTN-формализму), сетевым графиком [22] и др.

Теперь рассмотрим конкретизацию всех компонентов базовой модели ТПП ($ТПП$), исходя из приведенной выше модели M_{proto} . В этом случае множество параметров $T = \{T_i \mid i \in \mathbb{N}\}$ будет определяться конкретными элементами из модели архитектуры $M_{ИЭС}$ (НФ-операции, формализованные операции, накопители и др.). Множество условий ТПП представляется четверкой $C = \langle Frag, F_{check}, R_{frag}, Data \rangle$, где $Frag = \{Frag_i \mid i \in \mathbb{N}\}$ - множество так называемых *фрагментов ТПП*, т.е. неделимых частей ТПП; $R_{frag} : T \times Frag \rightarrow \{0, 1\}$ - функция, определяющая, попадает ли заданный параметр из T в конкретный фрагмент ТПП; функция F_{check} показывает возможность использования каждого элемента иерархии РДПД из $M_{ИЭС}$ в качестве конкретного параметра из T ; $Data$ - множество потоков данных между элементами из T .

Здесь важно отметить, что исходя из накопленного опыта разработки прикладных ИЭС, на структуру ТПП накладывается определенное ограничение - в любой ТПП содержится хотя бы один *обязательный* фрагмент и произвольное число *опциональных* фрагментов, связанных с обязательным фрагментом по крайней мере одним потоком данных. Как правило, первый элемент из множества $Frag$ включает в себя НФ-операцию, поскольку именно она играет основную роль в построении архитектуры текущего прототипа ИЭС [1].

Компонент L модели ТПП представляется в виде *абстрактной сети задач*, которая при подстановке конкретных значений параметров T позволяет получить конкретные задачи плана разработки, т.е. $L = \langle A_{templ}, R_{abstr-prec}, F_{incl}, F_{concr} \rangle$, где A_{templ} - множество паттернов задач, из которых могут быть получены плановые задачи при подстановке конкретных значений

параметров; $R_{templ-prec}$ - функция, определяющая отношение предшествования между элементами из A_{templ} ; $F_{incl} : A_{templ} \rightarrow \{0, 1\}$ - функция, принимающая значение 1, если для реализации данной ТПП при заданном наборе параметров необходимо выполнение данной задачи; $F_{concr} : A_{templ} \rightarrow A_{atom}$ - функция получения конкретных плановых задач из паттернов A_{templ} .

В целом, описанная выше совокупность отдельных компонентов модели интеллектуальной среды и их конкретизация представляет собой необходимый базис для реализации разработанного метода планирования действий инженера по знаниям.

2. Метод планирования действий инженеров по знаниям для построения прототипа ИЭС

Как было показано выше, назначение разработанного подхода заключается в генерации плана действий инженера по знаниям для построения прототипов прикладных ИЭС на основе модели архитектуры $M_{ИЭС}$, технологических знаний (декларативных в виде множества ТПП и операционных в виде множества ПИК). Рассмотрим предложенный метод планирования действий инженера по знаниям, опираясь на общую постановку задачи, введенные ограничения и рабочие определения.

2.1. Постановка задачи

Для описания постановки задачи $PlanTask_{ИЭС}$ используем общую формальную постановку задачи планирования в пространстве состояний, приведенную в [10, 11], где используется такое базовое понятие, как система помеченных переходов (Labeled Transition System) или система дискретного случая, которая представляется в виде $\Sigma = \langle S, A, E, \gamma \rangle$, где S - множество состояний; A - множество действий; E - множество событий; $\gamma : S \times (A \cup E) \Rightarrow 2^S$ - функция перехода. Таким образом, для системы Σ , с начальным состоянием s_0 и подмножеством целевых состояний S_g требуется найти последовательность действий, соответствующих переходам между состояниями $\{s_0, s_1, s_2, \dots\}$ таких, что $s_1 \in \gamma(s_0, a_1)$, $s_2 \in \gamma(s_1, a_2)$, ..., $s_k \in \gamma(s_{k-1}, a_k)$ и $s_k \in S_g$.

В качестве исходных данных выступают: модель архитектуры $M_{ИЭС}$, описанная с помощью иерархии РДПД; технологическая БЗ (мо-

дель WKB), содержащая множество ТПП (модель TPP) и ПИК (модель PIK). Кроме того, вводятся следующие ограничения и рабочие определения.

1. Из состава модели архитектуры $M_{ИЭС}$, построенной на этапе анализа системных требований, используется только множество элементов и множество потоков данных, представимых в виде *размеченного ориентированного* графа $G_{РИЛ}$, где метки определяют связь между элементами иерархии РДПД и вершинами, и дугами графа. Данный граф называется *обобщенной* РДПД, и он может быть однозначно получен из исходной модели архитектуры $M_{ИЭС}$.

2. *Фрагмент* обобщенной РДПД представляет собой произвольный *связанный подграф*, содержащийся в $G_{РИЛ}$. *Экземпляром* ТПП называется совокупность TPP и фрагмента обобщенной РДПД, удовлетворяющего условиям применимости (компонент C модели TPP) соответствующей ТПП. *Покрытием* ($Cover$) обобщенной РДПД является множество экземпляров ТПП, с взаимно непересекающимися фрагментами, содержащими все вершины $G_{РИЛ}$ (или *покрывают* весь $G_{РИЛ}$).

3. *Грубым* покрытием обобщенной РДПД называется такое покрытие РДПД, в котором все экземпляры ТПП содержат только обязательные фрагменты. *Точное* покрытие является расширением *грубого* покрытия за счет включения в него опциональных фрагментов.

4. Включению каждого *фрагмента* ТПП в покрытие сопоставляется некоторая стоимость, определяемая экспертной оценкой на основе технологического опыта, что концептуально соответствует усредненным затратам человеческих ресурсов на реализацию соответствующего фрагмента модели архитектуры ИЭС.

Таким образом, задачу генерации плана разработки прототипа ИЭС, с учетом исходных данных и введенных ограничений, удобно представить в терминах состояний и переходов в виде модели $PlanTask_{ИЭС} = \langle S_{ИЭС}, A_{ИЭС}, \gamma, Cost, s_0, G_{ИЭС}, F_{COVER} \rangle$, где $S_{ИЭС}$ - множество состояний графа $G_{РИЛ}$, описывающих текущее покрытие $Cover$; $A_{ИЭС}$ - множество возможных действий над $G_{РИЛ}$, заключающихся в добавлении к покрытию фрагментов конкретных экземпляров ТПП (полное множество формируется в совокупности WKB и $G_{РИЛ}$); γ - функция перехода между состояниями; $Cost$ - функция, опреде-

ляющая стоимость последовательности переходов; s_0 - начальное состояние, описывающее пустое покрытие; $G_{ИЭС}$ - функция определения принадлежности состояния к целевому; F_{COVER} - функция генерации плана разработки ($Plan$) из покрытия ($Cover$).

Решением $PlanTask_{ИЭС}$ является план действий инженера по знаниям (модель $Plan$), который может быть однозначно получен путем преобразования последовательности переходов, связанных с генерацией покрытия $a_1 \dots a_n$ ($a_i \subset A_{ИЭС}, n \in \mathbb{N}$), приводящих к целевому состоянию s_G (такого, что $G(s_G) = I$) за минимальную среди альтернатив стоимость, определяемую элементом $Cost$. Таким образом, план должен быть оптимальным, для чего необходимо использовать допустимую эвристическую функцию (описана в Разделе 2.3).

2.2. Описание метода генерации плана действий инженера по знаниям

Метод генерации плана условно может быть разбит на четыре этапа: получение обобщенной РДПД ($G_{РИЛ}$) из модели архитектуры $M_{ИЭС}$; генерация точного покрытия $Cover$ с помощью эвристического поиска; генерация плана действий инженера по знаниям ($Plan$) на основе полученного детального покрытия ($Cover$); генерация представления плана (PR) на основе покрытия ($Cover$).

На первом этапе для получения *обобщенной РДПД* ($G_{РИЛ}$) из модели архитектуры $M_{ИЭС}$ используется алгоритм рекурсивной детализации, функционирующий следующим образом. На начальном шаге инициализируется пустой граф $G_{РИЛ}$, и из $M_{ИЭС}$ выбирается диаграмма верхнего уровня и входящие в нее элементы (НФ-операции, накопители и др.), которые условно разделяются на две группы - детализируемые и недетализируемые (детализируемым элементом считается элемент РДПД, которому соответствует диаграмма более нижнего уровня). На следующем шаге для всех недетализируемых элементов в $M_{ИЭС}$ создаются вершины в $G_{РИЛ}$ с соответствующими метками, а детализируемые элементы заменяются на содержимое детализирующих РДПД более нижнего уровня. Данный шаг выполняется рекурсивно до тех пор, пока не будут обработаны все детализируемые элементы, после чего создаются дуги $G_{РИЛ}$ с соответствующими метками на основе потоков данных $M_{ИЭС}$.

На втором этапе производится генерация *точного покрытия* как состояния $s_G \subset S_{IES}$ графа $G_{РИЛ}$, путем эвристического поиска в пространстве состояний S_{IES} , а именно – детерминированного планирования в пространстве состояний. Поскольку при генерации используется «стоимость» добавления новых фрагментов к текущему покрытию (определяемая элементом $Cost$), то осуществляется *оптимальное* планирование (описание допустимой эвристической функции, позволяющей выполнять генерацию оптимальных планов, приводится в разделе 2.3.).

Иллюстрация промежуточного состояния, полученного в процессе данного этапа, приводится на Рис. 1. Здесь в покрытии участвуют две разных ТПП ($SDP1$ и $SDP2$), при этом присутствует два экземпляра SPI . Элементы $Op1$, $NF1$ и $E3$ в совокупности удовлетворяют условиям применимости (компонент C модели $ТПП$) обязательного фрагмента $SDP2$ (обведен сплошной линией), поэтому данные элементы могут быть покрыты соответствующим фрагментом при подстановке $Op1$, $NF1$ и $E3$ в параметры ТПП (компонент T модели $ТПП$). Накопитель $S2$ удовлетворяет условиям применимости опционального фрагмента $SDP2$, поэтому может быть добавлен к соответствующему экземпляру $SDP2$. Опциональные фрагменты на иллюстрации обведены пунктирной линией.

Следует отметить, что в общем случае существует множество точных покрытий заданной обобщенной РДПД, отличающихся стоимостью, что связано, прежде всего, с содержанием технологической БЗ. Например, в определенных ситуациях один и тот же подграф $G_{РИЛ}$ может быть покрыт экземпляром одной ТПП, либо несколькими экземплярами других ТПП, суммарная стоимость которых отличается. Как правило, подобные ситуации в технологической БЗ разрешаются по принципу «частное – предпочтительней», поэтому при генерации покрытия будет выбрана одна ТПП с более низкой стоимостью.

На третьем этапе на основе полученного покрытия с помощью функции F_{COVER} осуществляется генерация плановых задач и элементов модели $Plan$. Совокупность экземпляров ТПП, содержащихся в полученном покрытии, формирует множество глобальных задач из плана действий инженера по знаниям (элемент A_G модели $Plan$). В процессе детализации из каждого

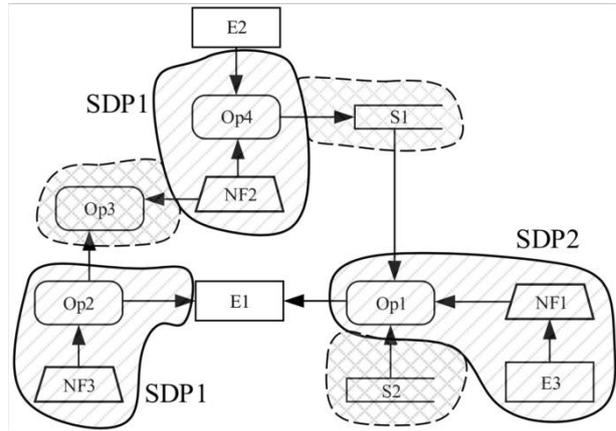


Рис. 1. Грубое и точное покрытие обобщенной РДПД

фрагмента (опционального или обязательного) генерируется подмножество задач, составляющих в совокупности A_{atom} , а на основе дуг обобщенной РДПД строится отношение R_{prec} (описывая тем самым сеть задач [22]).

На финальном этапе формируется представление плана (элемент PR модели $Plan$), которое предназначено для удобства отображения плана инженеру по знаниям (по умолчанию используется иерархическое представление). Для генерации формируется двухуровневое дерево, на первом уровне которого содержатся группы, соответствующие глобальным задачам из A_G , а на втором – плановые задачи, полученные преобразованием включенных в покрытие фрагментов ТПП. Последовательность выполнения задач определяется с помощью алгоритма *топологической сортировки* [25].

2.3. Описание специализированной эвристической функции для сокращения пространства поиска

Как показано в [24] и других работах, пространство поиска может быть существенно сокращено за счет использования проблемно-ориентированной эвристической функции вместо универсальных. В данном случае для определения эвристической функции был проведен анализ накопленного опыта прототипирования ИЭС, на основе чего были выделены следующие стандартные свойства технологической БЗ (компонент $WKВ$): обязательный фрагмент любой ТПП содержит одну и только одну НФ-операцию; опциональные фрагменты любой ТПП не содержат НФ-операций; стоимость

реализации любого фрагмента ТПП строго больше нуля. Исходя из этого, обозначим с помощью C_{NF} минимальную стоимость *обязательного* фрагмента, а C_F - минимальную стоимость *опционального* фрагмента среди всех ТПП, содержащихся в *WKB*.

Рассмотрим произвольное состояние s , полученное в процессе генерации покрытия (*Cover*) для обобщенной РДПД ($G_{РДПД}$), которое требуется оценить с помощью эвристической оценки. Обозначим N - количество вершин $G_{РДПД}$, соответствующих НФ-операциям, N_{COVER} - количество покрытых вершин, соответствующих НФ-операциям, а M_{COVER} - количество вершин, относящихся к формализованным операциям и накопителям, покрытых *Cover*. Тогда $M = G_{РДПД}$. $U - N$ - суммарное количество вершин обобщенной РДПД, соответствующее накопителям и формализованным операциям.

Эвристическая функция произвольного состояния s (содержащего покрытые *Cover*), оценивает минимально возможную стоимость оптимального плана до целевого состояния и вычисляется следующим образом: $h^{AT}(s) = (N - N_{COVER}) C_{NF} + (M - M_{COVER})/M * C_F$. При этом если функция $h^{AT}(s)$ является допустимой, то это даст возможность находить *оптимальные планы*. Пример вычисления значения эвристической функции приводится на Рис. 2, где покрытие в текущем состоянии содержит один экземпляр ТПП с двумя опциональными и одним обязательным фрагментом. Пусть стоимость текущего покрытия эквивалентна 13. Обобщенная РДПД содержит две непокрытых НФ-операции, соответствующих элементам *NF1* и *NF3*. Предположительно, в технологической БЗ содержатся 5 различных ТПП (*SDP1...SDP5*), и минимальной стоимостью

обладает обязательный фрагмент *SDP2* ($cost = 7$), а минимальный по стоимости опциональный фрагмент находится в *SDP1* ($cost = 3$). Таким образом, значения $h^{AT}(s) = 2 * 7 + (4)/6 * 3 = 16$, что определяет «удаленность» заданного состояния от целевого.

Несложно доказать, что предложенная функция является допустимой, если технологическая БЗ обладает перечисленными выше свойствами.

3. Особенности реализации интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ

Текущая версия интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ [1-3] включает такие компоненты как ядро, подсистему пользовательского интерфейса и библиотеку расширений, реализующую взаимодействие с различными операционными ПИК. Подсистема пользовательского интерфейса обладает удобным графическим интерфейсом, на основе которого обеспечивается взаимодействие ПИК с инженером по знаниям с помощью экранных форм. Технологическая БЗ условно подразделяется на библиотеку расширений, хранящую операционные знания в виде плагинов, реализующих соответствующие операционные ПИК, и на декларативную часть. В *ядре* реализуется вся основная функциональность автоматизированной поддержки разработки прототипов ИЭС, управление файлами проекта, управление расширениями и др.

Интеллектуальный планировщик является частью ядра и реализует функциональность, связанную с планированием процессов прототипирования ИЭС (Рис. 3.). Рассмотрим кратко особенности функционирования новой версии планировщика, для чего удобно выделить операционные компоненты (прямоугольники), информационные компоненты (овалы), а также потоки данных между ними (стрелки).

С помощью препроцессора иерархии РДПД производится преобработка иерархии РДПД за счет ее преобразования в одну обобщенную диаграмму максимальной детализации (несмотря на то, что инженеру по знаниям удобнее строить подобные диаграммы в виде иерархии, для автоматизированного анализа более эффективной является форма в виде одной диаграммы).

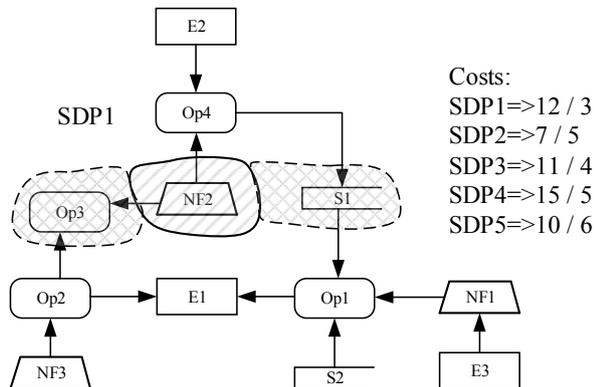


Рис. 2. Пример вычисления значения эвристической функции

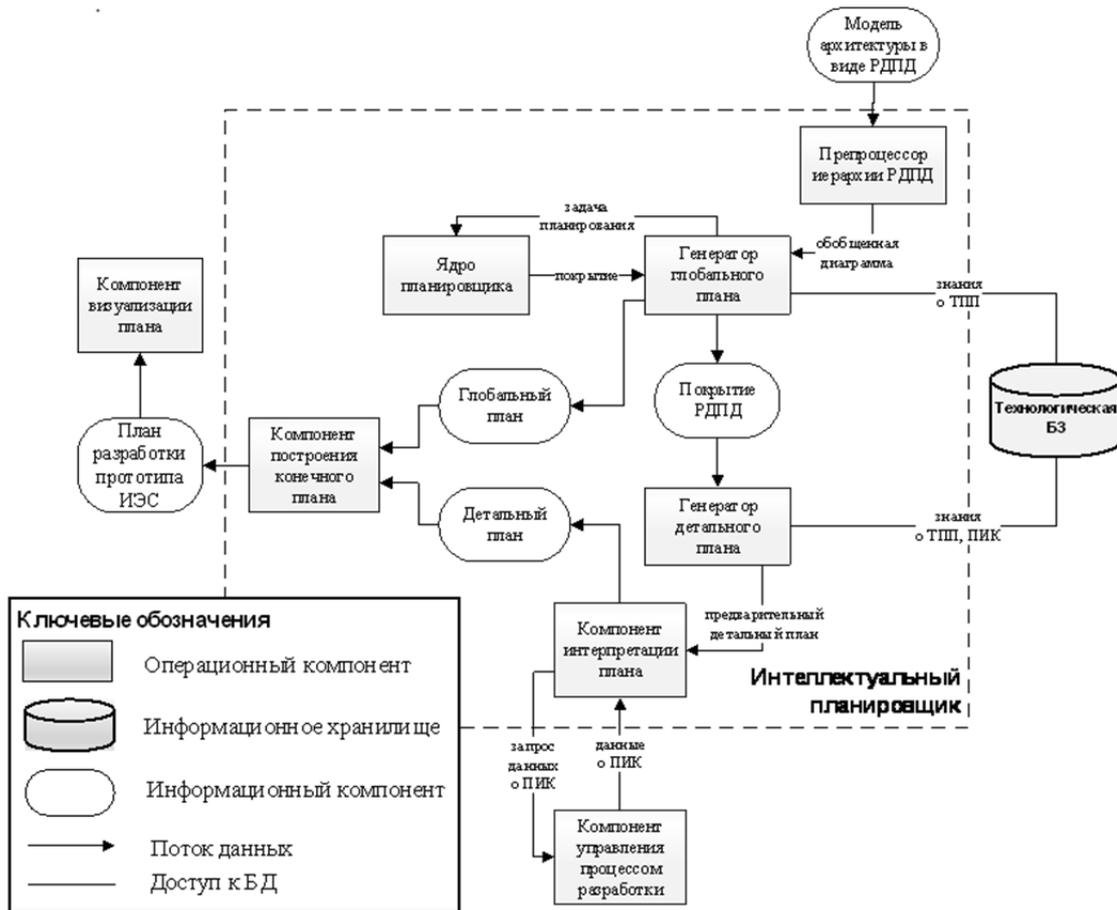


Рис. 3. Архитектура интеллектуального планировщика

Задача покрытия детализированной РДПД имеющимися ТПП реализуется с помощью генератора глобального плана, который на основе технологических знаний из БЗ и построенной обобщенной РДПД обеспечивает выполнение задачи на основе описанного выше метода, в результате чего строится *точное покрытие*, преобразуемое в дальнейшем в *глобальный план разработки*.

Генератор детального плана на основе полученного покрытия РДПД и технологических знаний строит детализацию каждого элемента покрытия, формируя тем самым предварительный детальный план. Затем на основе анализа доступных ПИК и их версий (данные о которых запрашиваются у компонента управления процессом разработки), компонентом интерпретации плана формируется *детальный план*, в котором каждая задача связана с конкретным ПИК и может быть выполнена инженером по знаниям.

С помощью компонента построения конечного плана производится формирование *представления* плана разработки в виде, например,

иерархической двухуровневой структуры, которое может быть использовано другими компонентами интеллектуальной программной среды, например, визуализировано с помощью компонента визуализации плана.

Реализация выполнена на языке C++11. Использованы среда разработки Visual Studio 2012, 2013, 2015 и Qt Creator. Графических интерфейсов основан на фреймворке Qt 5.7. Приблизительный объем программного обеспечения интеллектуального планировщика насчитывает суммарно около 30 тыс. строк кода.

4. Использование интеллектуальной программной среды для разработки прототипа ИЭС

Рассмотрим особенности применения интеллектуальной программной среды на примере разработки прототипа динамической ИЭС для управления медицинскими силами и средствами при крупных дорожно-транспортных про-

исшествиях (по заказу Всероссийского Центра Медицины Катастроф «Защита» [21]).

В качестве базовых задач в процессе прототипирования динамической ИЭС рассматривались такие задачи, как: мониторинг конкретного участка федеральной трассы и сигнализация о дорожно-транспортном происшествии (ДТП), включая первичную оценку типа ДТП; планирование мероприятий по устранению последствий ДТП; диагностика пострадавших (определение типа травм и степени тяжести); планирование мероприятий по госпитализации пострадавших в конкретное лечебное учреждение; мониторинг процесса госпитализации и состояния пострадавших.

Первым этапом ЖЦ автоматизированного построения ИЭС на основе задачно-ориентированной методологии является этап анализа системных требований пользователя, на котором происходит построение модели архитектуры ($M_{ИЭС}$) разрабатываемой ИЭС, т.е. по сути, осуществляется идентификация решаемой проблемы. Основываясь на классификации ИЭС «с точки зрения архитектуры», разрабатываемую в данном случае систему можно отнести к ИЭС с «глубинной» интеграцией компонентов [1], где средний уровень интеграции соответствует спецификациям наборов функций в *модели архитектуры* ИЭС. Определение спецификаций осуществляется инженером по знаниям, в результате чего строится $M_{РИД}$ в виде иерархии расширенных диаграмм потоков данных (РДПД) [1], являющейся основой $M_{ИЭС}$.

После завершения этапа построения $M_{ИЭС}$ прототипа ИЭС все дальнейшие функции инженера по знаниям, связанные с планированием разработки текущего прототипа ИЭС, реализуются с помощью *интеллектуального планировщика* на основе технологической БЗ, включающей в себя совокупность ТПП и ПИК. При этом новая версия интеллектуального планировщика позволяет осуществлять генерацию плана независимо от сложности модели архитектуры, в частности, количества НФ-операций и др.

На этапе инициализации интеллектуальным планировщиком производится анализ технологической БЗ на предмет наличия доступных ТПП и ПИК. Следует отметить, что в зависимости от доступности отдельных ПИК, некоторые фрагменты конкретных ТПП могут быть нереализуемы в текущей конфигурации.

Далее интеллектуальным планировщиком осуществляется построение *обобщенной РДПД* (Рис. 4) с помощью алгоритма рекурсивной детализации на основе исходной модели архитектуры ($M_{ИЭС}$) текущего прототипа, на которой присутствуют четыре НФ-операции (Планировать госпитализацию пострадавших, Диагностировать пострадавших, Планировать устранение последствий ДТП, Диагностировать первично тип ДТП).

Затем интеллектуальным планировщиком производится построение *грубого покрытия*, т.е. непересекающихся *обязательных* фрагментов ТПП из технологической БЗ. Построение грубого покрытия осуществляется на основе описанного выше метода с использованием специальной эвристической оценки для сокращения пространства поиска.

На следующем этапе строится *детальное покрытие*, т.е. расширение грубого покрытия *опциональными* фрагментами ТПП. Например, ТПП, связанная с НФ-операцией «Планировать госпитализацию пострадавших», расширяется фрагментами, связанными с построением имитационной модели, интеграцией с БД, интеграцией с ГИС, построением объяснений и др.

Затем из *детального покрытия* интеллектуальным планировщиком осуществляется генерация плана разработки прототипа ИЭС в два этапа. На первом этапе строится *глобальный план*, т.е. укрупненный сетевой график, где отношение предшествования определяется на основе анализа потоков данных обобщенной РДПД. На втором этапе каждый укрупненный фрагмент преобразуется в сеть элементарных задач, набор которых определяется конкретной ТПП и набором используемых опциональных фрагментов, получая тем самым *детальный план* (Рис. 5).

Например, для экземпляра ТПП «Построение динамических ИЭС» элемент глобального плана раскрывается одним *обязательным* и пятью *опциональными* фрагментами и содержит в общей сложности 17 плановых задач. В итоге получается сеть *детального плана*, т.е. сеть, состоящая из элементарных плановых задач. Следует отметить, что предусмотрено формирование с помощью интеллектуального планировщика визуального *представления плана*, после чего инженер по знаниям может приступить к выполнению плана.

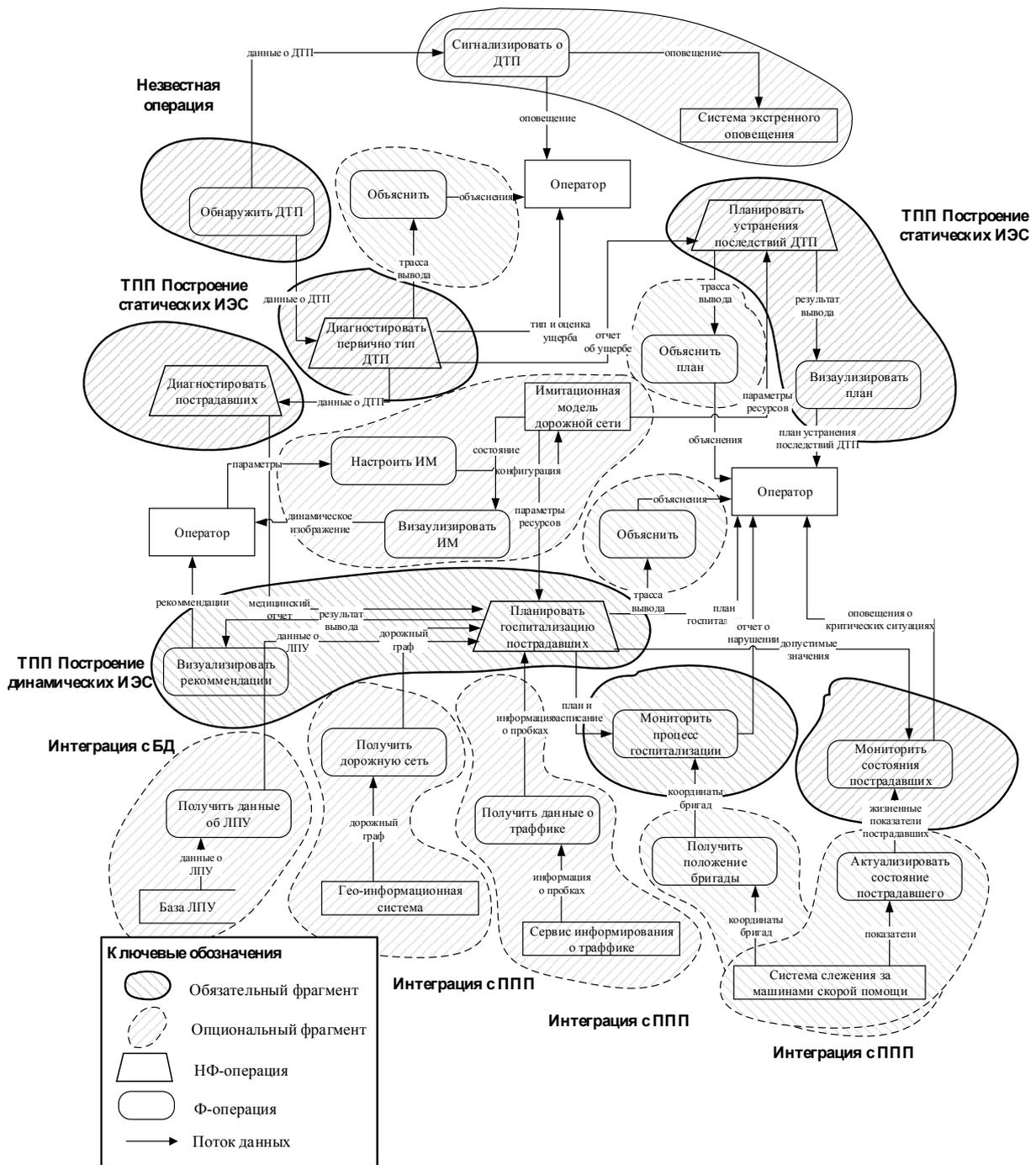


Рис. 4. Обобщенная РДПД, грубое и точное покрытие для прототипа ИЭС

Процесс выполнения плановой задачи может быть инициирован инженером по знаниям с помощью графического интерфейса интеллектуального планировщика. Затем для выполнения плановой задачи производится поиск

соответствующего инструментального ПИК, которому передается управление по специальному программному интерфейсу. После завершения задачи инженер по знаниям может отметить данную задачу как завершенную.

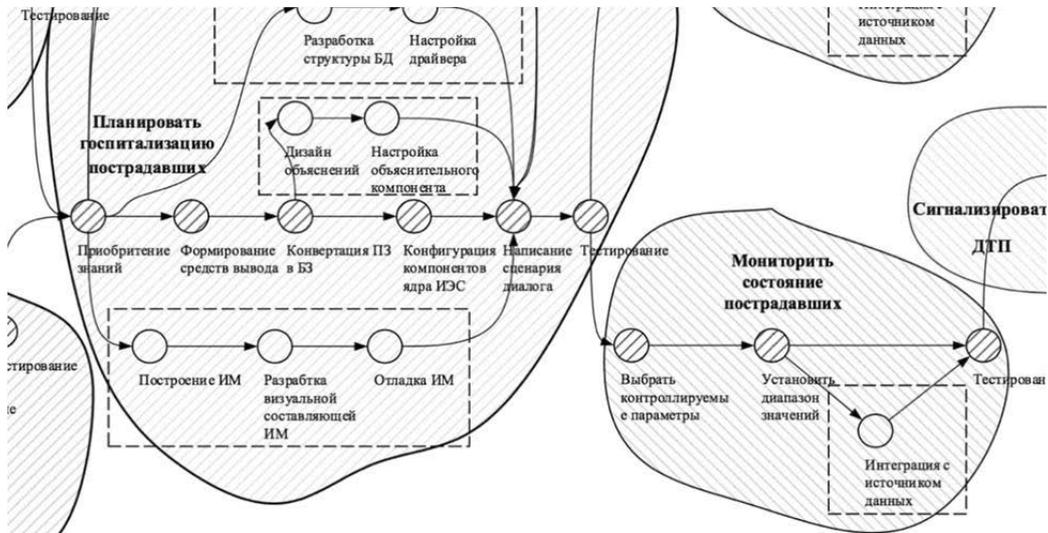


Рис. 5. Фрагмент детального плана построения прототипа ИЭС

Заключение

В рамках приведенных исследований была разработана новая технология построения прототипов прикладных ИЭС на основе использования интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ. Здесь существенная роль отводится интеллектуальному планировщику, с помощью которого минимизируются риски ошибочных действий инженеров по знаниям (аналитиков знаний), сокращаются стоимостно-временные показатели разработки на каждом этапе жизненного цикла прототипирования ИЭС и т.д. Данная технология может эффективно применяться для построения широкого класса ИЭС различной архитектурной типологии. В настоящее время проводятся исследования и разработки, связанные с дальнейшим развитием базовых компонентов интеллектуальной программной среды комплекса АТ-ТЕХНОЛОГИЯ, а также их экспериментальная апробация путем создания прототипов динамических ИЭС для интеллектуального управления сложными техническими объектами (ТПП «Построение динамических ИЭС») и разработкой обучающих ИЭС и веб-ИЭС, функционирующих на основе обобщенной онтологии «Интеллектуальные системы и технологии» (ТПП «Построение обучающих ИЭС»).

Литература

1. Рыбина Г.В. Теория и технология построения интегрированных экспертных систем. – М.: Научтехлитиздат, 2008. – 482 с.

2. Рыбина Г. В., Блохин Ю. М. Методы и средства интеллектуального планирования: применение для управления процессами построения интегрированных экспертных систем // Искусственный интеллект и принятие решений. 2015. №1. С. 75–93.
3. Rybina G.V., Blokhin Y.M. Use of intelligent planning for integrated expert systems development // 8th IEEE International Conference on Intelligent Systems, IS 2016, Sofia, Bulgaria, September 4-6, 2016. Pp. 295–300.
4. Гаврилова Т.А., Лещева И.А. Системный взгляд на подготовку инженеров по знаниям и бизнес аналитиков // Пятнадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2016 (3-7 октября 2016г., Смоленск, Россия). Труды конференции. В 3-х томах. Т.1. – Смоленск: Универсум, 2016. С. 16-23.
5. Грибова В.В., Клещев А.С., Крылов Д.А., Москаленко Ф.М., Смагин С.В., Тимченко В.А. "Проект IACPaas. Комплекс для интеллектуальных систем на основе облачных вычислений" // Искусственный интеллект и принятие решений. 2011. №1. С. 27-35.
6. Грибова В.В., Клещев А.С., Крылов Д.А., Москаленко Ф.М. и др. Базовая технология разработки интеллектуальных сервисов на облачной платформе IACPaas. Часть 1. Разработка базы знаний и решателя задач // Программная инженерия. 2015. №12. С. 3-11.
7. Хорошевский В. Ф. PIES-технология и инструментарий PIES Workbench для разработки систем, основанных на знаниях // Новости искусственного интеллекта. 1995. №2. С. 7-64.
8. Sacerodi E.D. The nonlinear nature of plans // James Allen James Hendler and Austin Tate, editors. Readings in Planning. – Morgan Kaufman, 1990. P. 162-170.
9. Осипов Г. С. Методы искусственного интеллекта. – М. : ФИЗМАТЛИТ, 2011. – 296 с.
10. Nau D.S. Current trends in automated planning // AI Magazine 28(4), 2007. P. 43–58
11. Ghallab M., Nau D.S., Traverso P. Automated planning - theory and practice. - Elsevier, 2004. - 663 p.
12. Рассел С., Нордвиг П. Искусственный интеллект: современный подход. 2-е изд. Пер. с англ. - М. : Издательский дом «Вильямс», 2006. - 1408 с.

13. Ghallab M., Dana S. Nau, Traverso P. *Automated Planning and Acting*, 1st Edition. - Cambridge: Cambridge University Press, 2016. - 368 p.
14. Яковлев К.С., Макаров Д.А., Баскин Е.С. Метод автоматического планирования траектории беспилотного летательного аппарата в условиях ограничений на динамику полета // Искусственный интеллект и принятие решений. 2014. №4. С.3-17.
15. Zou G., Chen Y., Xu Y. et al. Towards automated choreographing of web services using planning // Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence. 2012. P. 178–184.
16. Garrido A., Morales L., Serina I. Using AI planning to enhance E-Learning processes // Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012. - AAAI Press, 2012. - P. 47–55.
17. Parkinson S., Longstaff A., Crampton A., Gregory P. The application of automated planning to machine tool calibration // Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012. - AAAI Press, 2012. - P. 216–224.
18. Бурков В. Н., Коргин Н. А., Новиков Д. А. Введение в теорию управления организационными системами/ Учебник Под ред. Д. А. Новикова. - М.: Книжный дом «ЛИБРОКОМ», 2009 - 264 с.
19. Бурков В.Н., Буркова И.В., Губко М.В., Динова Н.И., Еналеев А.К., Кондратьев В.В., Коргин Н.А., Новиков Д.А., Цветков А.В., Чхартишвили А.Г., Щепкин А.В. Механизмы управления: Управление организацией: планирование, организация, стимулирование, контроль/ Учебное пособие. Изд. 2-е, перераб. и доп.- М.: ЛЕНАНД, 2013. -216 с.
20. Coles A. J. Opportunistic branched plans to maximise utility in the presence of resource uncertainty // ECAI 2012 - 20th European Conference on Artificial Intelligence. Including Prestigious Applications of Artificial Intelligence (PAIS-2012) System Demonstrations Track. – IOS Press, 2012. - P. 252–257.
21. Рыбина Г. В., Блохин Ю. М., Шилкин И. П. Особенности прототипирования динамических интегрированных экспертных систем с использованием интеллектуального планировщика комплекса АТ-ТЕХНОЛОГИЯ // Интегрированные модели и мягкие вычисления в искусственном интеллекте. Сборник научных трудов VIII-й Международной научно-технической конференции (Коломна, 18-20 мая 2015 г.). В 2-х томах. Т.2. - М. : Физматлит. 2015. С. 677–683.
22. Коротков М.А. Методы сетевого планирования. – М.: ИПУ РАН, 2003. – 16 с.
23. Erol K., Hendler J. A., Nau D. S. HTN planning: Complexity and expressivity // Proceedings of the 12th National Conference on Artificial Intelligence, Seattle, WA, USA, July 31 - August 4, 1994, Vol. 2. - 1994. Pp. 1123–1128.
24. Korf R., Felner A. Recent Progress in Heuristic Search: A Case Study of the Four-Peg Towers of Hanoi Problem // Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007. - AAAI Press, 2007. – P. 2324-2329.
25. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C. *Introduction to Algorithms* (3. ed.). — MIT Press. 2009. – 1312 p.
26. G2 Platform: <http://www.gensym.com>
27. SHINE Expert System: https://en.wikipedia.org/wiki/SHINE_Expert_System
28. Environmental software, planning & management: <http://www.ess.co.at/RTXPS/>

Рыбина Галина Валентиновна. Профессор кафедры «Кибернетика» Национального исследовательского ядерного университета «МИФИ» (НИЯУ МИФИ). Окончила Московский инженерно-физический институт (государственный университет) в 1971 году. Доктор технических наук. Количество печатных работ: около 500. Область научных интересов: интеллектуальные системы и технологии, статические, динамические и интегрированные экспертные системы, интеллектуальные диалоговые системы, многоагентные системы, инструментальные средства. E-mail: galina@ailab.mephi.ru

Блохин Юрий Михайлович. Ассистент кафедры «Кибернетика» Национального исследовательского ядерного университета «МИФИ» (НИЯУ МИФИ). Окончил НИЯУ МИФИ в 2011 году. Количество печатных работ: около 30. Область научных интересов: интеллектуальные системы и технологии, интеллектуальное планирование, интегрированные экспертные системы (статические, динамические обучающие), эвристический поиск, инструментальные средства.

Methods and software of intelligent planning for integrated expert systems construction

G.V. Rybina, Y.M. Blokhin

This work is focused on topical problem of development of efficient software tools for intelligent software development, in particular integrated expert systems (IES), which have powerful functionality and scalable architecture. Automated planning methods are used to increase automation level of IES development processes at designing and requirements analysis stages. General problems of development of modern instrument basis for IES construction are discussed. In this context basic concept of intelligent software environment of AT-TECHNOLOGY workbench for IES development basing on problem-oriented methodology is given. Planning problem of prototyping process is given which is base for implementation of basic components of intelligent software environment – intelligent planner and technological knowledge base. The method for automation based on heuristic search is proposed. It uses reusable components and standard design procedures which are part of technological knowledge base. An

example of using intelligent program environment for development dynamic IES prototype for control and planning of medical resources during large traffic accidents.

Keywords: artificial intelligence, integrated expert systems, problem-oriented methodology, intelligent program environment model, AT-TECHNOLOGY workbench, automated planning.

References

1. Rybina G.V. Teoriya i tehnologiya postroeniya integrirovannih ekspertnih system. – M.: Nauchtehlitizdat, 2008. – 482 p.
2. Rybina G.V., Blokhin Y.M. Metody i sredstva intellektual'nogo planirovaniya: primeneniye dlya upravleniya processami postroeniya integrirovannykh ehkspertnykh sistem // *Iskusstvennyy intellekt i prinyatie reshenij*. 2015. №1. P. 75–93.
3. Rybina G.V., Blokhin Y.M. Use of intelligent planning for integrated expert systems development // 8th IEEE International Conference on Intelligent Systems, IS 2016, Sofia, Bulgaria, September 4-6, 2016. Pp. 295–300.
4. Gavrilova T.A., Leshcheva I.A. Sistemnyy vzglyad na podgotovku inzhenerov po znaniyam i biznes analitikov // *Pyatnadcataya nacional'naya konferenciya po iskusstvennomu intellektu s mezhdunarodnykh uchastiem KII-2016 (3-7 oktyabrya 2016g., Smolensk, Rossiya)*. Trudy konferencii. V 3-h tomah. T.1. – Smolensk: Universum, 2016. P. 16-23.
5. Gribova V.V., Kleshchev A.S., Krylov D.A., Moskalenko F.M., Smagin S.V., Timchenko V.A. "Proekt IACPaaS. Kompleks dlya intellektual'nykh sistem na osnove oblachnykh vychislenij" // *Iskusstvennyy intellekt i prinyatie reshenij*. 2011. №1. P. 27-35.
6. Gribova V.V., Kleshchev A.S., Krylov D.A., Moskalenko F.M. i dr. Bazovaya tekhnologiya razrabotki intellektual'nykh servisov na oblachnoy platforme IACPaaS. CHast' 1. Razrabotka bazy znaniy i reshatelya zadach // *Programmnyaya inzheneriya*. 2015. №12. P. 3-11.
7. Horoshevskiy V. F. PIES-tekhnologiya i instrumentarij PIES Workbench dlya razrabotki sistem, osnovannykh na znaniyah // *Novosti iskusstvennogo intellekta*. 1995. №2. P. 7-64.
8. Sacerodi E.D. The nonlinear nature of plans // James Allen James Hendler and Austin Tate, editors. *Readings in Planning*. – Morgan Kaufman, 1990. P. 162-170.
9. Osipov G. S. Metody iskusstvennogo intellekta. - M. : FIZMATLIT, 2011. - 296 p.
10. Nau D.S. Current trends in automated planning // *AI Magazine* 28(4), 2007. P. 43–58
11. Ghallab M., Nau D.S., Traverso P. *Automated planning - theory and practice*. - Elsevier, 2004. - 663 p.
12. Rassel S., Nordvig P. *Iskusstvennyy intellekt: sovremennyy podhod, 2-e izd.*: Per. s angl. - M. : Izdatel'skiy dom «Vil'yams», 2006. - 1408 p.
13. Ghallab M., Dana S. Nau, Traverso P. *Automated Planning and Acting, 1st Edition*. - Cambridge: Cambridge University Press, 2016. - 368 p.
14. Yakovlev K.S., Makarov D.A., Baskin E.S. Metod avtomaticheskogo planirovaniya traektorii bespilotnogo letatel'nogo apparata v usloviyah ogranichenij na dinamiku poleta // *Iskusstvennyy intellekt i prinyatie reshenij*. 2014. №4. P.3-17.
15. Zou G., Chen Y., Xu Y. et al. Towards automated choreographing of web services using planning // *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*. 2012. P. 178–184.
16. Garrido A., Morales L., Serina I. Using AI planning to enhance E-Learning processes // *Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012*. - AAAI Press, 2012. - P. 47–55.
17. Parkinson S., Longstaff A., Crampton A., Gregory P. The application of automated planning to machine tool calibration // *Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling, ICAPS 2012*. - AAAI Press, 2012. - P. 216–224.
18. Burkov V.N., Korgin N.A., Novikov D.A. *Vvedeniye v teoriyu upravleniya organizatsionnymi sistemami: Uchebnyk / Pod red. D. A. Novikova*. - M.: Knizhnyy dom «LIBROKOM», 2009 - 264 p.
19. Burkov V.N., Burkova I.V., Gubko M.V., Dinova N.I., Enaleev A.K., Kondrat'ev V.V., Korgin N.A., Novikov D.A., Cvetkov A.V., CHkhartishvili A.G., SHCHepkin A.V. *Mekhanizmy upravleniya: Upravlenie organizatsij: planirovaniye, organizatsiya, stimulirovaniye, kontrol': uchebnoye posobie*. Izd. 2-e, pererab. i dop. M.: LENAND, 2013. -216 p.
20. Coles A. J. Opportunistic branched plans to maximise utility in the presence of resource uncertainty // *ECAI 2012 - 20th European Conference on Artificial Intelligence. Including Prestigious Applications of Artificial Intelligence (PAIS-2012) System Demonstrations Track*. – IOS Press, 2012. - P. 252–257.
21. Rybina G.V., Blohin Y.M., Shilkin I.P. Osobennosti prototipirovaniya dina- micheskih integrirovannykh ehkspertnykh sistem s ispol'zovaniem intellektual'nogo planirovshchika kompleksa AT-TEKHNOLOGIYA // *Integrirovannyye modeli i myagkie vychisleniya v iskusstvennom intellekte. Sbornik nauchnykh trudov VIII-j Mezhdunarodnoy nauchno-tekhnicheskoy konferencii (Kolomna, 18-20 maya 2015 g.)*. V 2-h tomah. T.2. - M.: Fizmatlit. 2015. P. 677–683.
22. Korotkov M.A. *Metody setevogo planirovaniya*. – M.: IPU RAN, 2003. – 16 p.
23. Erol K., Hendler J. A., Nau D. S. HTN planning: Complexity and expressivity // *Proceedings of the 12th National Conference on Artificial Intelligence, Seattle, WA, USA, July 31 - August 4, 1994, Vol. 2*. - 1994. P. 1123–1128.
24. Korf R., Felner A. Recent Progress in Heuristic Search: A Case Study of the Four-Peg Towers of Hanoi Problem // *Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*. - AAAI Press, 2007. – P. 2324-2329.
25. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C. *Introduction to Algorithms (3. ed.)*. — MIT Press. 2009. – 1312 p.
26. G2 Platform: <http://www.gensym.com>
27. SHINE Expert System: https://en.wikipedia.org/wiki/SHINE_Expert_System
28. Environmental software, planning & management: <http://www.ess.co.at/RTXPS/>

Rybina Galina. Professor of «Cybernetics» department of National Research Nuclear University “MEPhI” (Moscow, Kashirskoe sh., 31), doctor of technical science, author of more than 500 scientific papers. Main interests – intelligent systems and technologies, static, dynamic and integrated expert systems, intelligent dialogue systems, multiagent systems, software tools. E-mail: galina@ailab.mephi.ru

Blokhin Yuri. Assistant of «Cybernetics» department of National Research Nuclear University “MEPhI” (Moscow, Kashirskoe sh., 31). Author of approximately 30 scientific papers. Main interests – intelligent systems and technologies, automated planning, integrated expert systems (static, dynamic, tutoring), heuristic search, software tools.