

- ного технического университета Украины. Машиностроение. — К., 2000. — Вып. 38, Т. 2. — С. 205–209.
2. Щодро А. Е. Кинематика потока и водообмен за косорасположенными донными уступами гидротехнических сооружений / А. Е. Щодро // Гидравлика и гидротехника : респ. межвед. научно-техн. сборник. — К. : Техника, 1978. — Вып. 26. — С. 59–64.
 3. Щодро О. С. Порівняння розрахункових та експериментальних даних про кінематичну структуру, поле тисків, наносотранспортуючу та розмивну спроможності потоку в зоні просторової відривної течії за кося розташованими грядами та донними уступами / О. С. Щодро // Вісник НУВГП : збірник наукових праць. — Рівне : НУВГП, 2006. — Вип. 4 (36), Ч. 1. — С. 173–179.

Given are results of study of the intensity of channel stream near the bottom ridge of sediments, depending on the angle of its location and height. The study is based on numerical simulation of the bottom resistance in field of the ridge. It was established that at the height of the ridge equal to 1/3 the depth of the stream most powerful deformations can occur in its location at an angle of 35–40 degrees to the oncoming flow.

Key words: *numerical simulation, intensity of the water flow, local erosion, flow over the ridge.*

Отримано: 31.03.2014

УДК 004.81

И. А. Чмырь, д-р техн. наук

Одесский государственный экологический университет, г. Одесса

МОДЕЛИРОВАНИЕ АКТИВНОГО АГЕНТА ЭРОТЕТИЧЕСКОГО ДИАЛОГА

Статья посвящена описанию модели активного агента диалогового процесса эротетического типа. Отличительной особенностью предлагаемой модели является то, что в ее основу положена идея диалоговой базы знаний. Диалоговая база знаний определяется как совокупность диалогового метода доступа к памяти вопросов и самой памяти вопросов. В статье приведен необходимый онтологический базис, в рамках которого описана общая модель активного агента, и даталогическая модель диалогового метода доступа. Приведен пример отображения модели диалогового метода доступа в логическую модель реляционной базы данных.

Ключевые слова: *эротетический диалог, диалоговая база знаний, диалоговый метод доступа, плохо-формализуемые задачи.*

Введение. Разработка полной и точной модели естественного диалогового процесса или разговора, достаточной для синтеза универсального диалогового агента является постоянным вызовом для тех, кто ставит своей целью создание робота, способного поддерживать

полномасштабный естественно-языковой разговор с партнером-человеком. Этот вызов подогревается научно-фантастической прозой и фильмами, где упомянутая проблема считается решенной. Естественный диалоговый процесс не подвержен никаким ограничениям. Он не ограничен ни предметной областью разговора, ни правилами поведения его участников. Анализ протоколов естественных диалогов показывает, что в разговоре возможна спонтанная смена ролей участников диалога, спонтанная смена темы разговора, спонтанный переход диалогового процесса в монолог одного из участников и т.д. Понятно, что создание полной и точной модели естественного диалога, достаточной для синтеза универсального диалогового агента, эквивалентно созданию полной и точной модели ментальной деятельности человека.

Всегда ли искусственный диалоговый агент должен уметь поддерживать полномасштабный естественный диалоговый процесс? Если иметь в виду научную фантастику, то ответ должен быть положительным, однако если иметь в виду конкретные прикладные системы искусственного интеллекта, то ответ может быть и отрицательным. Известны примеры, когда достижение цели диалога осуществляется в ходе диалогового процесса, в котором участники соблюдают ряд ограничений. В *эротетическом диалоге* одним из главных ограничений является вопросно-ответный статус информационных сообщений, которыми обмениваются разговаривающие. Диалогом эротетического типа пользуется, например, Сократ и, с его помощью, успешно реализует, изобретенный им метод убеждений. Диалогом эротетического типа, как правило, пользуется следователь при допросе подозреваемого. И Сократ, и следователь используют эротетический диалог, не только для коммуникации, но и в качестве метода решения задачи.

В общем случае, диалог эротетического типа можно рассматривать как средство решения задач, относящихся к классу плохо-формализуемых (ill-formalized) [1]. Множество примеров плохо-формализуемых задач можно найти в области автоматизированного проектирования. Например, при разработке программ теплофизического проектирования, как правило, основные усилия направлены на создание расчетного ядра программы, поскольку здесь можно опереться на хорошо разработанный математический аппарат и на традиционные технологии программирования. Однако, расчетное ядро «окружает» множество плохо формализуемых задач. Рассмотрим процесс теплофизического проектирования элемента конструкции энергетической установки. Расчетный блок в такой системе используется для определения полей температур или термических напряжений с помощью метода сеток или метода конечных элементов. В зависимости от производительности компьютера и количества используемых элементов этот процесс продолжается единицы или десятки минут, в то время как под-

готовка сетевой модели и данных, необходимых для расчета, занимает месяцы работы эксперта. Не менее трудоёмок процесс интерпретации экспертом полученных массивов чисел, т.е. переход от полей температур или напряжений к конкретным конструкторским решениям. Использование компьютера, как на начальном этапе проектирования, так и на этапе интерпретации расчётных результатов предполагает умение решать перечисленные плохо формализуемые задачи.

Целью статьи является описание модели искусственного диалогового агента, способного поддерживать целенаправленный эротетический диалог. Предполагается, что сферой применимости такого диалогового агента являются системы, ориентированные на решение плохо-формализуемых задач в области интеллектуальных систем обучения и автоматизированного проектирования.

Необходимый онтологический базис. Назовём участников диалогового взаимодействия *диалоговыми агентами*. Диалоговый агент исполняет одну из двух ролей: *активного агента* или *реактивного агента*. Агент становится активным, когда ему необходимы знания, которые он предполагает получить от своего партнёра, и реактивным — в том случае, когда он предоставляет своему партнёру, имеющиеся у него знания. Анализ протоколов диалоговых процессов позволяет сделать вывод о том, что инициатором смены ролей является реактивный агент. Активный агент удовлетворён своей ролью и, после смены ролей, старается восстановить прежнюю роль при первой же возможности.

Диалоговый агент ассоциирован с одной или двумя личностями. В том случае, когда диалоговое взаимодействие осуществляется между двумя личностями, имеет место *внешний диалог*. Диалоговое взаимодействие может осуществляться в виде «разговора с самим собой». В этом случае роль активного и реактивного агентов попеременно исполняет одна и та же личность. Такой диалог назовём *внутренним диалогом*. Вербальный внутренний диалог часто называют монологом, а ментальный — размышлением.

В случае внешнего диалога, знания, которые запрашивает активный агент, хранятся в долговременной памяти реактивного агента, а знания, на которые ссылается реактивный агент в своём ответе — в долговременной памяти активного агента. В случае внутреннего диалога и запрашиваемые и получаемые знания хранятся в долговременной памяти одной и той же личности.

Внешнее проявление диалога, или его протокол, представляет собой цепь *информационных транзакций*. Характерной особенностью диалога, которая отличает его от других форм интерактивного взаимодействия, например, от работы с прикладной системой баз данных, является наличие логической связи не только между компонентами транзакции, но и между самими транзакциями. Именно на-

личие такой логической связи превращает набор транзакций в целенаправленный интеллектуальный процесс.

Транзакция представлена двумя *информационными посылками*. В естественном диалоге информационная посылка активного агента *имеет статус вопроса*. Слово «статус» означает, что информационная посылка активного агента не обязательно является вопросом в лингвистическом смысле, и может принимать различные формы. Для активного агента важным является получение доступа к запрашиваемым знаниям, а не форма запроса, при помощи которого, осуществляется этот доступ. Информационные посылки активного агента могут варьироваться от простого «да/нет» вопроса, до продолжительного внутреннего диалога. Поэтому уместно говорить о степени интеррогативности информационной посылки активного агента. Информационную посылку активного агента, часто, можно понимать как *поисковое предписание*, необходимое для получения доступа к знаниям реактивного агента. Информационная посылка реактивного агента диалога имеет статус ответа. Эту информационную посылку можно также понимать как *поисковое предписание*, необходимое для получения доступа к знаниям, активного агента.

В эротетическом диалоге информационная посылка активного агента представляет собой *явный ли-вопрос*, либо *какой-вопрос*, а информационная посылка реактивного агента — прямой ответ на этот вопрос [2]. В эротетическом диалоге, знания, которые запрашивает активный агент и которые возвращает реактивный агент, являются декларативными. Поэтому, модель структуры эротетической транзакции, в контексте представления знаний, должна основываться на некоторой идее представления декларативных знаний. Система декларативных знаний человека, по всей видимости, едина, но эротетическая транзакция имеет дело с небольшой порцией этой системы. Более полную онтологию естественного и эротетического диалоговых процессов можно найти в [3].

Активный диалоговый агент. В целенаправленном диалоговом процессе, с помощью которого осуществляется решение плохо-формализуемой задачи, ключевую роль играет активный агент, поскольку именно он является носителем метода решения задачи.

Введем понятие «диалоговая база знаний», которую будем обозначать *DiKB*. Диалоговая база знаний является частью модели активного диалогового агента и может рассматриваться либо как память, хранящая знания, необходимые для поддержки целенаправленного диалогового процесса, либо как память, в которой находятся знания о методе решения задачи. В знаниях, находящихся в *DiKB* выделим декларативную и процедурную части. Следуя концепции унифицированной когнитивной модели АСТ-R, предложенной Джоном Андерсоном [4], будем придерживаться идеи раздельного хранения процедурных и декларативных знаний в *DiKB* и представим ее структуру так, как показано на рис. 1.

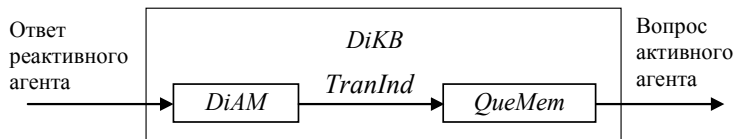


Рис. 1. Структура диалоговой базы знаний активного агента.

DiAM — диалоговый метод доступа к памяти вопросов.

TranInd — имя последующей транзакции. *QueMem* — память вопросов

С точки зрения процедурно-декларативной дихотомии знаний, порции декларативных знаний, передаваемых от активного агента к реактивному, в диалоговых транзакциях, хранятся в памяти вопросов *QueMem*. Память *QueMem* будем рассматривать как память с произвольным доступом к запоминающим/адресуемым элементам, которые проиндексированы именами диалоговых транзакций. Получив имя очередной диалоговой транзакции *TranInd*, память вопросов возвращает спецификации вопроса активного агента на данной транзакции. Логическая структура вопроса активного агента является темой отдельного исследования. Для целей настоящей статьи достаточно принять, что спецификации вопроса существуют. Процедурные знания активного агента, это знания, которые можно представить продукцией вида: «ЕСЛИ <ответ>, ТО <имя транзакции>». Они хранятся в памяти, названной *DiAM* (Dialogue Access Method), поскольку по отношению к *QueMem* эта память работает как метод доступа к адресуемым элементам *QueMem*. *DiAM* трансформирует текущий ответ реактивного агента в имя последующей транзакции. Однако, мы будем рассматривать *DiAM* в более широком смысле, и считать, что *DiAM*, с одной стороны, трансформирует ответ реактивного агента в имя транзакции, а с другой — является хранилищем метода решения плохо формализуемой задачи.

Диалоговая база знаний является диалоговым компонентом модели активного агента. Для того, чтобы диалог мог осуществляться как процесс, кроме диалогового компонента необходим алгоритмический компонент. Алгоритмический компонент, взаимодействуя с *DiKB*, формирует вопрос из его спецификаций и презентует его реактивному агенту, а также воспринимает ответ реактивного агента и формирует релевантную ему имя последующей транзакции. Выделим в алгоритмическом компоненте часть, которая единообразно/унифицировано обрабатывает любую транзакцию. Унифицированность здесь означает, во-первых, что для реализации любой транзакции необходимо выполнить одну и ту же последовательность действий, а во-вторых, что эта последовательность действий не зависит от задачи, решаемой в процессе диалога. Назовем эту часть «циклом диалогового метода» и обозначим как *DiMC*. Для того, чтобы цикл диалогового метода был унифицированным и пригодным для обработки любой транзакции он должен отражать унифицированную когнитивную мо-

дель восприятия, адаптированную к условиям эротетического диалогового процесса. В качестве такой модели предлагается рассматривать циклическую модель восприятия Найсера, и основанную на ней когнитивную модель диалогового цикла [5; 6].

Среди деятельностей, выполняемых *DiMC*, есть две деятельности, требующие уточнения. Во-первых, это деятельность, при помощи которой осуществляет формирование вопроса и, во-вторых — деятельность, при помощи которой осуществляется обработка ответа и определение имени следующей транзакции. Поскольку *DiMC* — это унифицированный процесс, то, как формирование вопроса, так и обработка ответа должны быть стандартизованы. Иными словами, *DiMC* должен уметь стандартным образом формировать любой вопрос и обрабатывать любой ответ. Ясно, что даже при достаточно обширном стандарте могут встретиться случаи, когда стандартных средств недостаточно. Например, при генерации вопроса, в том случае, когда он представлен не вербально, невозможно в его спецификации учесть все многообразие невербальной презентации информации. Необходимость в нестандартных средствах становится особенно очевидной, когда при обработке ответа необходимо выполнить специфическую численную обработку. Поэтому, кроме базового процесса *DiMC*, целесообразно использовать специализированные и «внешние», по отношению к *DiMC*, процессы. Эти процессы должны вызываться из *DiMC* и возвращать ему управление после завершения работы. В [7] такие процессы получили наименование *процессы-демоны*. Таким образом, в общем случае, в алгоритмической части модели должна присутствовать библиотека специализированных «демонов» *TranDemon*, которые могут осуществлять «дообработку» транзакции в том случае, когда стандартных средств *DiMC* недостаточно. На рис. 2 приведена структура активного агента эротетического диалога, учитывающая, как даталогический, так и алгоритмический компоненты.

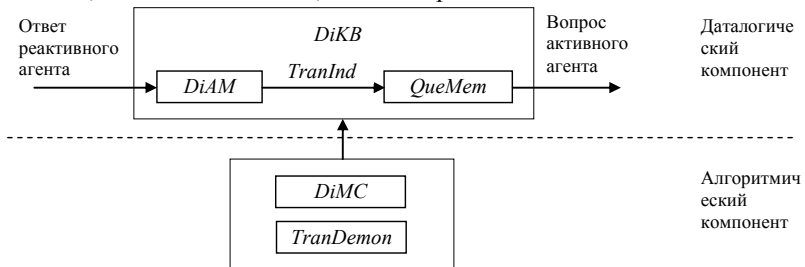


Рис. 2. Модель активного агента эротетического диалогового процесса. *TranDemon* — библиотека «демонов» транзакций. *DiMC* — цикл диалогового метода

С точки зрения процедурно-декларативной дихотомии знаний блок *DiMC* хранит (в алгоритмической форме) процедурные знания о

последовательности действий, необходимых для обработки любой диалоговой транзакции, а библиотека *TranDemon* — процедурные знания о действиях, которые дополняют деятельности *DiMC* при обработке некоторых специфических транзакций.

Сетевая модель диалогового метода доступа. Одним из наиболее распространенных способов представления процедурных знаний являются продукционные правила. Хотя процедурные знания человека не дискретны, а связаны в единое целое, базы знаний большинства экспертных систем представляют собой наборы отдельных, явно не связанных правил продукции. Хранение процедурных знаний в виде набора дискретных правил удобно с точки зрения их редактирования, однако порождает ряд проблем, среди которых наиболее заметной является проблема возникновения конфликтных ситуаций. Тем не менее, в литературе можно найти несколько способов *представления процедурных знаний в виде сети*, состоящей из явно связанных продукционных правил. Это, например: сеть вывода (inference net) [8]; дерево типа И/ИЛИ (AND/OR tree) [9]; дерево вывода (decision tree) [10]. Сетевое представление процедурных знаний обладает рядом преимуществ, среди которых отметим отсутствие конфликтных ситуаций, возможность совмещение процесса вывода с диалоговым процессом эротетического типа, и ряд других.

Будем считать, что *DiAM* имеет сетевую организацию и состоит из связанных между собой узловых элементов. Отдельный узел *DiAM* соответствует одной потенциальной диалоговой транзакции, а вся сеть может рассматриваться как *сценарий потенциальных протоколов диалогового процесса*. В каждый момент времени активным является только один узел, который соответствует текущей транзакции. Определение последующей транзакции и активизацию соответствующего узла осуществляет унифицированный процесс *DiMC* и, если это необходимо, то и демоны транзакции. Связи между узловыми элементами являются направленными и для каждого узла определяют множество возможных последующих транзакций.

Будем считать, что сеть *DiAM* неоднородна в том смысле, что она строится из узлов разных типов. Узлы отличаются назначением, описанием и правилом навигации. Идея неоднородности сети *DiAM* означает, во-первых, что сеть состоит из разнотипных узловых элементов, а во-вторых, что допустима различная типология этих элементов. В качестве примера рассмотрим сеть, состоящую из узловых элементов двух типов: (1) узел непосредственной навигации *Node1*, который соответствует транзакции, находящейся внутри сценария и предполагающей продолжение диалога и (2) терминальный узел *Node2*, которым завершается диалоговый процесс.

В последующем разделе сеть *DiAM* будет отображаться в схему реляционной базы данных. Для того, чтобы такое отображение было возможным, необходимо даталогическое представление каждого из введенных типов узлов. На рис. 3 приведено даталогическое представление узла типа *Node1*.

На рис. 3 использованы следующие обозначения: *TranInd* — имя транзакции; *NodeType* — тип узла, соответствующий данной транзакции; *QueInd* — имя вопроса активного агента, ассоциированного с данной транзакцией; *QueDemonInd* — имя процесса-демона, обслуживающего вопрос данной транзакции; *CurrAns[1]...CurrAns[r]* — текущие ожидаемые ответы реактивного агента; *AnsDemonInd* — имя процесса-демона, обслуживающего ответы данной транзакции; *NextTranInd* — имя следующей транзакции для ожидаемого ответа; *NNTranInd* — имя следующей транзакции для всех неожиданных ответов.

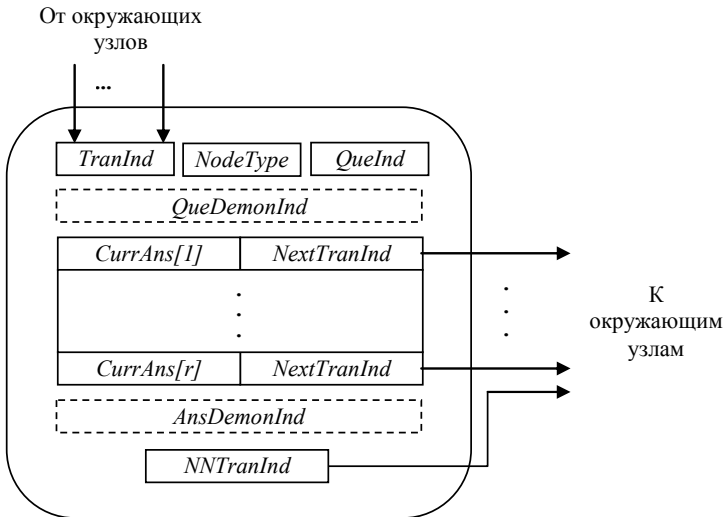


Рис. 3. Даталогическое представление узла типа *Node1*

Пунктиром отмечены необязательные описатели

Эротетический диалоговый процесс мы рассматриваем как целенаправленный процесс, осуществляющий решение плохо-формализуемой задачи. Для того, чтобы последовательность диалоговых транзакции могла рассматриваться как целенаправленный диалоговый процесс, любые две смежные транзакции должны быть связаны логикой решения задачи. Такая связь обеспечивается тем, что на поставленный вопрос активный агент ожидает не любой, произвольный ответ, а один из ожидаемых ответов *CurrAns[1]...CurrAns[r]*. Спецификой узла типа *Node1* является *непосредственная связь* каждого из ожидаемых ответов реак-

тивного агента с именем последующей транзакции *NextTranInd*. Все возможные ответы, которые не относятся к ожидаемым связаны с именем единственной последующей транзакцией *NNTranInd*.

Поскольку различным транзакциям могут соответствовать спецификации одного и того же вопроса, то среди описателей узла фигурирует имя вопроса *QueInd*. Дatalogическое представление узла типа *Node1*, приведенное на рис. 3, предполагает, что существуют два вида демонов транзакции *QueDemon* и *AnsDemon*, обслуживающие вопрос и ответ, соответственно. С учетом введенных обозначений правило навигации для узла типа *Node1*, в том случае, когда не задействованы демоны, можно записать в виде.

```

IF<распознан CurrAns[i]>
THEN <перейти к узлу NextNodeInd>
ELSE
IF <не распознан CurrAns[i]>
THEN <перейти к узлу NNodeInd>
    
```

(1)

На рис. 4. приведено дatalogическое представление узла сети типа *Node2*.

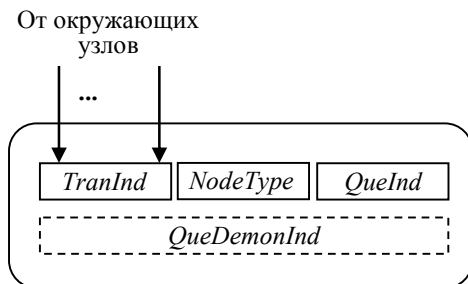


Рис. 4. Дatalogическое представление узла типа *Node2*

Поскольку узел типа *Node2* является целевым и переход на этот узел эквивалентен завершению работы диалогового метода, то, во-первых, его описание не предполагает перехода к последующим узлам, а, во-вторых, *QueInd* является именем *утверждения*, а не *вопроса*. При практической реализации диалогового сценария с методом доступа на основе сетевой модели, достижение узла типа *Node2*, часто, означает переход из *DiMC* в программную среду более высокого уровня, например, в среду операционной системы.

Отображение сетевой модели диалогового метода доступа в реляционную базу данных. Практическая реализация сети *DiAM*, в рамках дatalogического проектирования, может быть осуществлена путем ее отображения в логическую схему базу данных. Опишем сеть *DiAM*, состоящую из узловых элементов типа *Node1* и *Node2*, в виде

схемы абстрактной реляционной базы данных, используя нотацию, предложенную в [11].

Задача логического проектирования базы данных является, по сути, задачей информационного моделирования и не имеет единственного решения. Задача информационного моделирования сети *DiAM* заключается в описании сети в виде набора отношений. Ясно, что можно предложить несколько вариантов такого описания. Приведенная логическая модель реляционной базы данных не претендует на оптимальность, а, скорее, иллюстрирует конструктивность подхода.

Будем моделировать сеть *DiAM* следующими четырьмя отношениями.

$$DiAM = \{Tran, Navig, QueDemon, AnsDemon\}, \quad (2)$$

где *Tran* — отношение, моделирующее множество транзакций, связанных сетью *DiAM*; *Navig1* — отношение, моделирующее транзакции с узлами типа *Node1* и правилом навигации, описываемым формулой (1); *QueDemon* — отношение, моделирующее множество процессов-демонов, обслуживающих вопросы; *AnsDemon* — отношение, моделирующее множество процессов-демонов, обслуживающих ответы.

Следующим шагом в синтезе модели является представление каждого из отношений базы (2) в виде набора атрибутов. В приведенных ниже формулах атрибуты-ключи выделены подчеркиванием, а выражение $dom(Attr) = \{\dots\}$ специфицирует домен соответствующего атрибута.

$$Tran = \{\underline{TranInd}, NodeType, QueInd, NNTranInd\}. \quad (3)$$

Атрибут с именем *TranInd* моделирует имя транзакции. Имена транзакций различны. Домен этого атрибута состоит из *TranQ* имен, где *TranQ* — общее количество транзакций *DiAM*.

Атрибут с именем *NodeType* моделирует тип узлового элемента. Домен этого атрибута $dom(NodeType) = \{Node1, Node2\}$.

Атрибут с именем *QueInd* моделирует имя вопроса соответствующего транзакции. В различных транзакциях могут быть совпадающие имена вопросов. Домен этого атрибута состоит из *QueQ* имен, где *QueQ* общее количество различных вопросов в памяти вопросов. $TranQ \geq QueQ$.

Атрибут с именем *NNTranInd* моделирует имя транзакции, на которую осуществляется переход для случая неожиданного и, следовательно, нераспознаваемого ответа. Домен этого атрибута, также как и домен атрибута *TranInd* состоит из *TranQ* имен.

Каждая из транзакций *DiAM* описывается точно одним кортежем отношения *Tran*, поэтому количество кортежей отношения *Tran* равно *TranQ*.

$$Navig = \{\underline{TranInd}, \underline{CurrAns}, NextTranInd\}. \quad (4)$$

Атрибут с именем *TranInd* в отношении *Navig* имеет точно такой же смысл, как и в отношении *Tran*.

Атрибут с именем *CurrAns* моделирует распознаваемые ответы. Домен этого атрибута $dom(CurrAns) = \{CurrAns_1, \dots, CurrAns_r\}$, где r — наибольшее количество распознаваемых ответов для любой транзакции *DiAM*.

Атрибут с именем *NextTranInd* моделирует имя транзакции, на которую осуществляется переход для случая распознаваемого ответа. Домен атрибута состоит из *TranQ* имен.

Каждый распознаваемый ответ на каждой транзакции для узла с прямой навигацией описывается точно одним кортежем отношения *Navig1*. Количество кортежей отношения *Navig1* не превышает числа $r * TranQ$.

$$QueDemon = \{TranInd, QueDemonInd\}, \quad (5)$$

$$AnsDemon = \{TranInd, AnsDemonInd\}. \quad (6)$$

Отношения *QueDemon* и *AnsDemon* имеют простую структуру. Атрибуты этих отношений ставят в соответствие каждому имени транзакции, имя одного из процессов-демонов.

Выводы. В статье описана модель активного агента эротетического диалогового процесса, ориентированная на использование в прикладных системах искусственного интеллекта способных решать плохо-формализуемые задачи в ходе диалогового процесса. Модель легла в основу инструментальной системы «Процессор диалога», которая была разработана в 90-е годы в Одесском институте низкотемпературной техники и энергетики. При помощи программы Процессора диалога были сгенерированы ряд приложений в области интеллектуальных систем обучения и автоматизированного проектирования. Опыт применения инструментальной системы Процессора диалога позволяет сформулировать ряд методологических ограничений, в которых предлагаемая модель работает наиболее эффективно. Во-первых, декларативные и процедурные знания, которыми заполняется диалоговая база знаний, и которые хранят экспертные знания конкретного специалиста, должны отражать возможно большее количество вариантов развития диалогового процесса. Во-вторых, диалоговый процесс, в ходе которого осуществляется решение задачи, должен быть гармоничным. Гармоничным здесь назван такой диалоговый процесс, в котором диалоговые агенты удовлетворены распределением ролей и не стремятся их изменить.

Список использованной литературы:

1. Newell A. Remarks on the relationship between artificial intelligence and cognitive psychology / A. Newell // Theoretical Approaches to Non-Numerical Problem Solving / eds. R. Banerji, J. D. Mesarovich. — New York : Springer-Verlag, 1970. — P. 363–399.
2. Белнап Н. Логика вопросов и ответов / Н. Белнап, Т. Стил. — М. : Прогресс, 1981. — 288 с.

3. Чмырь И. А. Естественный диалог: моделирование диалоговой транзакции в контексте представления знаний / И. А. Чмырь // Московский государственный университет. Интеллектуальные системы. — 2012. — Т. 16. — Вып. 1–4. — С. 73–99.
4. Anderson J. R. Rules of the Mind / J. R. Anderson. — N.J. : Hillsdale, Lawrence Erlbaum Associates, 1993. — 410 p.
5. Neisser U. Cognition and Reality. Principles and implications of cognitive psychology / U. Neisser. — San Francisco : W. H. Freeman and Company, 1976. — 230 p.
6. Chimir I. A. Neisser's Cycle of Perception: Formal Representation and Practical Implementation / I. A. Chimir, W. A. Abu-Dawwas, M. A. Horney // Journal of Computer Science. — 2005. — Special issue on Intelligence, Communication and Computer Interaction. — P. 106–111.
7. Айтьян С. Х. Инструментальные средства разработки экспертных систем на ПЭВМ / С. Х. Айтьян, К. Р. Гуарян // Микропроцессорные средства и системы. — 1989. — № 6. — С. 55–61.
8. Winston P. H. Artificial Intelligence / P. H. Winston. — Addison-Wesley, 1984. — 359 p.
9. Попов Э. В. Экспертные системы / Э. В. Попов. — М. : Наука, 1987. — 211 с.
10. Levin R. I. A Comprehensive Guide to AI and Expert Systems: Turbo Pascal Edition / R. I. Levin, D. E. Drang, B. Edelson. — McGraw-Hill, 1988. — 182 p.
11. Мейер Д. Теория реляционных баз данных / Д. Мейер. — М. : Мир, 1987. — 331 с.

This paper describes the model of an active agent of erotetic dialogue process, based on the idea of a dialogue knowledge base. The dialogue knowledge base is defined as a set of a dialogue access method to the memory of questions and the memory of questions itself. The paper contains necessary ontological basis, which uses as a frame for description of a general model of an active agent and a datalogical model of the dialogue access method. The paper also contains an example of mapping of the datalogical model of the dialogue access method into a logical model of relational data base.

Key words: *erotetic dialogue, dialogue knowledge base, dialogue access method, ill-formalized problems.*

Отримано: 17.03.2014