

ПРИМЕНЕНИЕ СЕМАНТИЧЕСКОЙ ТЕХНОЛОГИИ КОМПОНЕНТНОГО ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ РЕШАТЕЛЕЙ ЗАДАЧ В ДИСТАНЦИОННОМ ОБУЧЕНИИ

С.С. Заливако, С.С. Старцев, Д.В. Шункевич

Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь, zalivako@mail.ru, ai@programist.ru, shu.dv@tut.by

Abstract. This article describes the intellectual problem solvers design technology application in distance learning. It also describes the general problem solver model, the library, which consists of compatible logical intellectual agents, the problem solvers design procedure and the infrastructure of technology. Considered one of the most important processes that accompany the solution of every problem - the explaining machine solution process. The main result of technology – using problem solvers in different intellectual reference systems.

Компьютерное обучение и самообучение является одной из самых активно развивающихся областей искусственного интеллекта. Включает в себя модели, методы и алгоритмы, ориентированные на автоматическое накопление и формирование знаний на основе анализа и обобщения данных, обучение по примерам (или индуктивное), а также традиционные подходы из теории распознавания образов [1]. В связи с этим возникает проблема решения задач из изучаемой пользователем предметной области. Семантическая технология компонентного проектирования интеллектуальных решателей задач направлена на упрощение процесса создания программных средств, поддерживающих решение задач вычислительного и невычислительного характера, сопровождающееся объяснением действий и принятых системой решений [2].

Опишем основные компоненты, которые входят в состав технологии:

- модель интеллектуального решателя задач;
- библиотека совместимых ip-компонентов (ip – intelligent property);
- система автоматизации проектирования;
- методика проектирования;
- help-система по проектированию решателей задач;
- инфраструктура технологии.

Модель решения задачи основана на многоагентном подходе, что позволяет использовать одних и тех же интеллектуальных агентов для решения разных задач и классов задач. В общем случае решатель задач представляет собой графодинамическую sc-машину (SC – semantic code), в состав которой входит ассоциативная перестраиваемая (графодинамическая) память (sc-память) и множество агентов [3].

Агенты представляют собой sc-операции. Для каждой операции существует условие инициирования – появление в памяти такой конструкции, существование которой позволяет операции начать свою работу. Операции сгруппированы в системы операций, которые обычно реализуют либо какую-то стратегию решения задачи, либо используют хранимые процедуры решения задач. Группировка операций в системы нацелена прежде всего на то, чтобы решать не какие-то конкретные задачи, а целые классы задач из какой-либо предметной области.

Такой подход к организации решения задач обеспечивает гибкость и расширяемость решателя путем добавления и удаления из его состава некоторой системы операций.

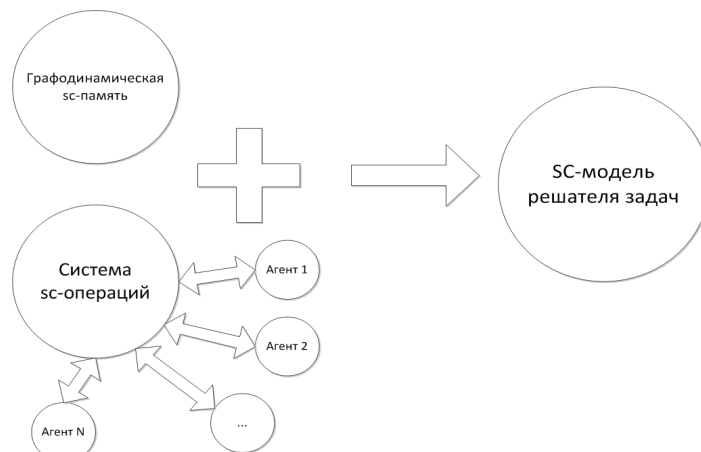


Рисунок 1 – Модель интеллектуального решателя задач

В результате анализа различных предметных областей было выделено 5 наиболее общих классов задач, которые описывают особенности в решении задач различных типов:

– **задачи на доказательство** – в рамках этих задач требуется установить истинность некоторого утверждения посредством посылок (утверждений, теорем, аксиом, результата решения похожих задач и т.п.);

– **задачи на преобразование** – данный класс задач подразумевает формулировку целевой ситуации, уточняющую желаемые вид и оптимизацию ответа (упростить, разложить на множители, проинтегрировать и т.п.);

– **задачи на описание** – это обобщение таких типов задач, как текстовые математические задачи с множеством неизвестных, в ходе решения которых используются методы решения систем алгебраических уравнений и неравенств, методы линейной оптимизации, методы динамического программирования, игровые и другие методы решения задач;

– **задачи на исследование** – в результате решения этого класса задач к имеющейся «картинке» добавляются какие-либо ценные факты, либо получаются следствия, требующие немедленного возвращения к внешней задаче, для которой предпринимается исследование. Таким образом, решение задачи происходит эволюционным путем («по спирали») [4];

– **задачи классификации** – в этом случае необходимо установить принадлежность объекта какому-то классу из заранее определенного списка посредством решающего правила или построению этого правила по имеющимся классифицированным объектам (обучение с учителем, дискриминантный анализ). Если же список классов не определен, то задача сводится к определению этого списка по имеющимся объектам (обучение без учителя, кластерный анализ).

Каждая задача в общем случае может и не относиться к какому-либо классу задач полностью, однако содержать элементы сразу нескольких классов. Соответственно, при решении таких задач сочетаются стратегии и модели, присущие нескольким классам задач.

Центральным элементом технологии является библиотека совместимых *ip*-компонент, которая имеет следующую структуру:

– **Библиотека готовых решателей задач** – включает в себя готовые наборы интеллектуальных логических агентов, которые могут обеспечить процесс решения задач в конкретных прикладных системах (по геометрии, теории графов, числовым системам, логике, русскому языку и др.);

– **Библиотека стратегий решения задач** – это наборы агентов, которые организывают решение задач в рамках определенной стратегии (поиск в глубину, поиск в ширину, сведение задачи к подзадаче, решение задачи с конца, метод «проб и ошибок» и др.);

– **Библиотека интеллектуальных логических агентов** – содержит агентов, которые находят ответы на вопросы пользователя или друг друга. Посредством этих агентов осуществляется процесс логического вывода, вычисление формул, обработка знаний и т.д.;

– **Библиотека процедур, написанных на различных языках программирования** – эта часть библиотеки схожа с библиотекой логических агентов за исключением того, что агенты не только логические и они могут быть платформенно зависимыми в силу их реализации на различных языках программирования;

– **Библиотека базовых преобразований** – это элементарные программы для работы с конструкциями в графодинамической памяти (поиск изоморфной конструкции по образцу, генерация изоморфной конструкции, поиск всех выходящих дуг, генерация узла, генерация дуги с атрибутом и др.).

Проектирование интеллектуальных решателей задач представляет собой достаточно трудоемкий процесс, поэтому была разработана задачно-ориентированная методика, содержащая четыре этапа:

1. создание тестового сборника задач, которые решаются в рамках исследуемой предметной области;

2. определение набора предметно независимых интеллектуальных агентов, которые будут использоваться при решении задач из тестового сборника;

3. специфицирования и разработка требований к каждому агенту, определенному разработчиком;

4. реализация и отладка программ на специальном языке SCP (Semantic Code Programming), ориентированном на обработку семантических сетей.

Таким образом, семантическая технология проектирования интеллектуальных решателей задач позволяет проектировать такие системы предметно независимых интеллектуальных агентов, которые способны решать унифицированным образом множество различных задач одного класса. Каждый из агентов представляет собой ip-компонент, который может быть использован в других прикладных системах. Многоагентная модель позволяет легко осуществлять интеграцию компонентов машин обработки знаний при условии корректной интеграции баз знаний.

Литература

1. Гаврилова Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский. – СПб.- Питер, 2000. – 384 с.

2. Заливако С.С., Старцев С.С., Савельева О.Ю., Шункевич Д.В. Семантическая технология компонентного проектирования интеллектуальных решателей задач / Заливако С.С. [и др.] // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». – Минск, БГУИР, 2011. – С. 265-286

3. Голенков В.В. Представление и обработка знаний в графодинамических ассоциативных машинах / Голенков В.В. [и др.]; под ред. В.В. Голенкова.- Минск, 2001.

4. Подколзин А.С. Компьютерное моделирование логических процессов. Архитектура и языки решателя задач. – М.: ФИЗМАТЛИТ, 2008. – 1024 с.