

УДК 004.896, 004.414.2

ОНТОЛОГО-ОРИЕНТИРОВАННОЕ ПРОЕКТИРОВАНИЕ ПОДСИСТЕМЫ ОЦЕНКИ ОБСТАНОВКИ ИНТЕЛЛЕКТУАЛЬНОГО АГЕНТА РЕАЛЬНОГО ВРЕМЕНИ

С.В. Лебедев (*lebedev.sv.etu@gmail.com*)
СПбГЭТУ «ЛЭТИ», Санкт-Петербург

М.Г. Пантелеев (*MPanteleyev@gmail.com*)
СПбГЭТУ «ЛЭТИ», Санкт-Петербург

Аннотация. Предложен подход к проектированию подсистем оценки обстановки беспилотных транспортных средств с высокой степенью автономности (БТС ВСА), основанный на использовании онтологических моделей предметных областей и универсального вычислительного каркаса. Подход является инвариантным к различным типам БТС ВСА и позволяет сократить время разработки программного кода и повысить его качество на этапе прототипирования новых и перспективных образцов БТС ВТА. Представлен пример формализации фрагмента предметной области и использования предложенного подхода.

Ключевые слова: подсистема оценки обстановки, интеллектуальный агент, онтология проектирования.

Введение

Развитие беспилотных транспортных средств (БТС) на современном этапе характеризуется двумя ключевыми тенденциями: динамичным расширением номенклатуры изделий, разрабатываемых для гражданских и военных применений, и повышением степени автономности за счет интеллектуализации бортовых алгоритмов. БТС с высокой степенью автономности (БТС ВСА) принято рассматривать в рамках концепции интеллектуальных агентов (ИА) [Gunetti et. al. 2010]. Разработка программного обеспечения (ПО) для таких систем является сложным трудоемким процессом, эффективность которого может быть повышена за счет использования средств автоматизации разработки ПО. Одним из ключевых требований к таким средствам, с учетом многообразия типов и конкретных образцов БТС, является обеспечение универсальности, т. е. поддержка процесса разработки ПО для БТС ВСА различного назначения,

функционирующих в различных средах. В последние годы активно развивается онтологический подход к автоматизации разработки и верификации ПО, позволяющий генерировать и настраивать ПО на конкретную предметную область (ПрО) с использованием онтологической модели соответствующей ПрО.

В настоящей работе рассматриваются теоретические основы и практические аспекты реализации онтологического подхода к проектированию одной из основных подсистем ИА — подсистемы оценки обстановки (ПОО). Подход позволяет сократить сроки и упростить процесс программной реализации ПОО на этапе прототипирования БТС ВСА различного назначения.

1 Место и функции ПОО в архитектуре ИА

В данной работе задача проектирования ПОО рассматривается в рамках обобщенной делиберативной архитектуры агента, представленной на рис. 1.

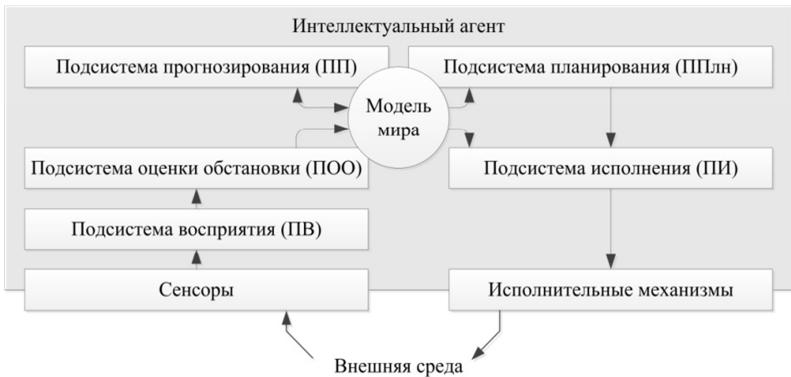


Рис. 1. Архитектура ИА РВ

Центральным компонентом данной архитектуры является модель мира (ММ), представляющая собой динамически формируемый агентом информационный образ среды, в которой он функционирует. Функция ПОО — построение и обновление ММ на основе текущих низкоуровневых сенсорных данных, поступающих от подсистемы восприятия (ПВ), их предыстории и априорных знаний.

Предполагается, что агент планирует и осуществляет целенаправленные действия в высокодинамичной многоагентной среде, в том числе в условиях активного противодействия. Таким образом, ММ содержит информационные образы объектов внешней среды, включая их

свойства (атрибуты) и отношения между ними.

Модели оценки обстановки активно разрабатываются в последние годы в рамках концепций слияния данных (Data Fusion) и ситуационной осведомленности (Situational Awareness).

Базовая модель ситуационной осведомленности [Endsley, 1995] разработана как когнитивная модель формирования ситуационной осведомленности у операторов сложных систем. Т.е. данная модель не ориентирована на разработку БТС ВСА.

Наиболее известной моделью слияния данных является JDL DF (Joint Directors of Laboratories Data Fusion) [Steinberg et al., 2009a]. Вопросы оценки обстановки с использованием данной модели рассмотрены, например, в [Steinberg, 2009b; Bowman et al., 2009]. Известные работы не ориентированы на решение проблем автоматизированной генерации программного обеспечения ПОО для конкретных БТС ВСА.

В [Kokar et al., 2009] рассмотрен основанный на ситуационной теории и модели Эндсли декларативный подход к созданию системы оценки обстановки. Подход использует онтологии для представления ситуаций и вывода новой информации с использованием дескриптивной логики и правил логического вывода. В отличие от данного подхода, предлагается совместить декларативный и процедурный подходы. Первый позволяет упростить работу со знаниями за счет специализированных редакторов и языков представления. Второй позволяет реализовать сложные алгоритмические решения.

В основе подхода лежит иерархическая модель потока вычисления элементов (объектов и отношений). На нулевом уровне на основе воспринятой и априорной информации вычисляются элементы первого уровня. Далее, на основе всей известной информации вычисляются элементы второго уровня и так далее до уровней элементов, требуемых для ППлн. В совокупности, полученная информация составляет ММ.

Такой подход позволяет сохранять вычислительные зависимости, обеспечивая возможность создания и обновления ММ при изменении элементов нижних уровней (в первую очередь, непосредственно воспринимаемых элементов).

2 Подход и базовая модель проектирования ПОО

2.1 Подход к проектированию

Предлагаемый подход к построению ПОО базируется на использовании инвариантного вычислительного каркаса и его онтологической настройки с учетом функционального назначения конкретного БТС ВСА, среды его функционирования, типов внешних

объектов, их существенных свойств и отношений.

В рамках данного подхода выделено два этапа проектирования:

1. Проектирование онтологической модели ПОО в соответствии с заданной ПрО, определяемой средой функционирования и функциональным назначением БТС ВСА;
2. Интеграция построенной онтологической модели в базовый вычислительный каркас.

Процесс проектирования при данном подходе может рассматриваться как отображение понятий ПрО на понятия вычислительного процесса ПОО. Данное отображение поддерживается рассмотренной ниже моделью вычислимости (см. п. 2.1). Сущности модели ПрО аннотируются сущностями модели вычислимости и полученная таким образом модель используется для построения модели процесса вычисления. На основе последней реализуется автоматизированная генерация программного кода.

Предполагается, что в подсистеме планирования реализован подход опережающего итеративного планирования, рассмотренный в [Пантелеев, 2012; Пантелеев, 2014].

2.2 Обобщенная модель ПОО

Обобщенная модель (ОМ) проектирования ПОО, соответствующая предложенному подходу, включает два компонента:

$$OM = \langle MPrO, MPP \rangle,$$

где MPrO – модель ПрО, MPP – модель процесса проектирования, формализующая структурные и функциональные аспекты ПОО.

MPP представлена тройкой:

$$MPP = \langle MP, MBч, MB \rangle,$$

где MP – модель представления, MBч – модель вычислимости, MB – модель вычисления.

MP предназначена для формализации понятий MPrO и включает следующие элементы:

$$MP = \langle O, P, GO, CO \rangle,$$

где O – множество объектов, P – множество отношений, GO – множество групповых объектов, где каждый объединяет группу объектов в рамках одного отношения, CO – множество системных объектов, где каждый объединяет группу объектов в рамках набора различных отношений (ролей). MP можно рассматривать как универсум для построения MM.

MB формализует процесс вычисления, схематично представленный на рис. 2, где L_k обозначает k-ый уровень вычисления (УрВ). Структура процесса вычисления динамически строится по мере получения новых восприятий и вычисления. Каждый УрВ содержит фрагмент MM, вычисленный на предыдущем уровне. На основе собственного фрагмента

и всех фрагментов ММ нижележащих уровней генератором создаются узлы вычисления (УзВ). УзВ вычисляет элемент ММ, записываемый в вышележащий уровень.

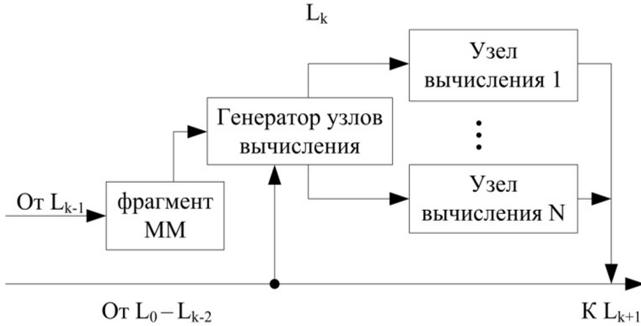


Рис. 2. Структура процесса вычисления

Для вычисления элемента к каждому УзВ присоединяется процедура, реализующая один из двух обобщенных шаблонов вычисления:

$$f_a : \langle o_1, o_2 \rangle \rightarrow r(o_1, o_2),$$

$$f_c : o_1 \rightarrow \langle o_2, r(o_1, o_2) \rangle.$$

Первый шаблон соответствует установлению отношения между двумя ранее вычисленными объектами. Второй – порождению нового объекта на основе ранее вычисленного объекта и априорно заданного между ними отношения.

Порождая новый элемент, УзВ связывает исходные элементы с порожденными, обеспечивая возможность реализации функции обновления при изменении исходных элементов.

МВч предназначена для отображения понятий МП на понятия МВ. Для этого используется понятие порождающего отношения, фиксирующего вычислительную зависимость между исходными и порождаемыми элементами. Само порождающее отношение однозначно связывается с УзВ.

Порождающие отношения позволяют отобразить единую МП на множество вариантов вычисления (например, когда один и тот же элемент может быть как воспринят, так и вычислен).

Гранулярность предложенной структуры вычисления позволяет компоновать в рамках каждой из итераций вычислительные задачи с учетом имеющихся ресурсов. Кроме того, для отдельных УрВ или УзВ может быть установлена приоритетность вычисления, отвечающая интересам подсистемы планирования.

2.3 Обобщенный метод проектирования ПОО

На основе предложенной ОМ сформулирован обобщенный метод проектирования ПОО, включающий следующие действия:

1. Формализовать МПрО в понятиях предложенной МП;
2. Задать порождающие отношения на данном множестве элементов ПрО, связывая исходные элементы с порождаемыми;
3. На основе полученных зависимостей породить структуру процесса вычисления, состоящую из узлов и уровней вычисления;
4. Реализовать и связать методы вычисления элементов с каждым из узлов вычисления;
5. На основе полученной модели сгенерировать программный код и интегрировать его в инвариантный каркас вычисления.

2.4 Онтология проектирования ПОО

На основе предложенной концептуальной модели в редакторе TopBraid Composer¹ была создана онтология проектирования ПОО. При ее создании использовались следующие спецификации: язык описания онтологий OWL²; язык запросов к онтологиям SPARQL³; язык, позволяющий связать понятия на языке OWL с запросами на языке SPARQL, – SPIN⁴. Выбранный редактор кроме средств описания, включает также средства выполнения SPARQL запросов.

Созданная онтология формализует на языке OWL понятия МПР, описанной в разделе 2.2. МП представлена следующими классами и подклассами: Entity, Object, GroupObject, SystemObject. Для представления отношений используются встроенные возможности языка OWL.

МВ представлена на рис. 3.

МВч представлена классами GenerativeRelation, Known, Percepted и Proposal. Known – родительский класс для представления априорно известных объектов, Proposal – класс для объектов-предложений порождающих отношений (см. раздел 2.4.1).

Класс GenerativeRelation связан с классом Entity отношениями gen_rel_base и gen_rel_derivative, фиксирующими соответственно исходные и порождаемые сущности.

¹ <http://www.topquadrant.com/tools/modeling-topbraid-composer-standard-edition/>

² <http://www.w3.org/TR/owl2-overview/>

³ <http://www.w3.org/TR/rdf-sparql-query/>

⁴ <http://spinrdf.org/>

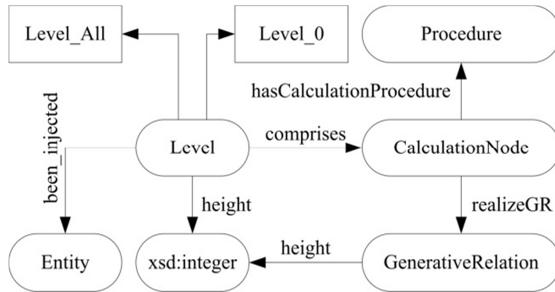


Рис. 3. Онтологическая модель вычислительного процесса

2.4.1 Автоматизация процесса проектирования на базе правил

В рамках предложенной онтологии были реализованы запросы на языке SPARQL с использованием SPIN нотации. Запросы предназначены для частичной автоматизации процесса проектирования: формирование вариантов (предложений) порождающих отношений на основе объектов и отношений ПО; порождение уровней и узлов вычисления на основе порождающих отношений.

Кроме того, с помощью правил может быть реализована проверка на предмет того, что все элементы модели ПрО могут быть вычислены.

2.4.2 Пример формализации

Рассмотрим простой фрагмент ПрО, онтологическая модель которого содержит классы «радар» (Р) и «батарея ПВО» (Б) и связывающее (Р) и (Б) отношение «выдавать целеуказание» (Ц).

Пример порождающего отношения – порождение (Б) на основе непосредственно воспринятого (Р). В этом случае нулевой УрВ ПОО включает в качестве фрагмента ММ (Р), а также УзВ для порождения (Б) и установления отношения (Ц) между (Р) и (Б). При этом (Б) будет относиться к фрагменту ММ первого УрВ. Соответствующие уровни и узел вычисления будут созданы в онтологии автоматически на основе данного порождающего отношения и соответствующего SPARQL запроса.

Другой пример порождающего отношения – установление отношения (Ц) между непосредственно воспринятыми (Р) и (Б). Соответствующий УзВ также будет включен в нулевой УрВ.

3 Заключение

Базовая онтология и основанный на ней подход позволяют упростить процесс проектирования и разработки ПОО БТС ВСА на этапе их прототипирования за счет автоматизации, использующей формальные представления артефактов проектирования и запросов на языке SPARQL.

Данный подход и онтология проектирования реализованы и апробированы при проектировании ПОО ИА, функционирующего в среде виртуального футбола.

В настоящее время ведется разработка средств генерации программного кода на заданном языке программирования на основе онтологии и средств его интеграции в инвариантный вычислительный каркас.

Список литературы

- [**Gunetti et. al. 2010**] Gunetti P., Dodd T., Thompson H. A software architecture for Autonomous UAV Mission Management and Control //Proceedings of the AIAA InfoTech@Aerospace Conference. – 2010.
- [**Bowman et al., 2009**] Bowman C. L., Steinberg A. N. Systems Engineering Approach for Implementing Data Fusion Systems/ Llinas J., Hall D. L., Liggins M. E. (ed.). Handbook of Multisensor Data Fusion: Theory and Practice. – CRC Press, 2009. – С. 561-596
- [**Endsley, 1995**] Endsley M. R. Toward a theory of situation awareness in dynamic systems //Human Factors: The Journal of the Human Factors and Ergonomics Society. – 1995. – Т. 37. – №. 1. – С. 32-64.
- [**Genesereth et al., 1987**] Genesereth M. R., Nilsson N. J. Logical Foundations of Artificial Intelligence. – Palo Alto: Morgan Kaufmann, 1987. – 406 с.
- [**Kokar et al., 2009**] Kokar, M.M., Matheus, C.J., Baclawski K. Ontology-based situation awareness //Information fusion. – 2009. – Т. 10. – № 1. – С. 83-98
- [**Steinberg et al., 2009a**] Steinberg A. N., Bowman C.L. Revisions to the JDL Data Fusion Model / Llinas J., Hall D. L., Liggins M. E. (ed.). Handbook of Multisensor Data Fusion: Theory and Practice. – CRC Press, 2009. – С. 44-67
- [**Steinberg, 2009b**] Steinberg A. N. Foundations of Situation and Threat Assessment / Llinas J., Hall D. L., Liggins M. E. (ed.). Handbook of Multisensor Data Fusion: Theory and Practice. – CRC Press, 2009. – С. 437-501
- [**Пантелеев, 2012**] Пантелеев М.Г. Концепция построения интеллектуальных агентов реального времени на основе модели опережающего итеративного планирования //Труды 12-ой национальной конф. по искусственному интеллекту с международным участием КИИ-2012. – Белгород: Изд-во БГТУ, 2012. – Т 3. – С.25-33.
- [**Пантелеев, 2014**] Пантелеев М.Г. Формальная модель опережающего итеративного планирования действий интеллектуальных агентов реального времени //Труды 14-ой национальной конф. по искусственному интеллекту с международным участием КИИ-2014. – Казань: Физматлит, 2014. – Т 1. – С.323-334.