

В.Б. Артеменко

Моделирование взаимодействия участников электронного обучения с использованием агент-ориентированного подхода

Рассмотрены направления создания агент-ориентированной модели, состоящей из динамически взаимодействующих трех типов участников электронного обучения: преподавателей – авторов дистанционных курсов, тьюторов и студентов. Описаны подходы к созданию упомянутой модели средствами *SWARM* – одного из наиболее популярных (некоммерческих) прикладных пакетов для построения агент-ориентированных моделей.

The directions are considered of creating the agent-oriented model which consists of three dynamically interacting types of participants of the electronic learning: the teachers-authors of the distance courses, tutors and students. The approaches to creation of such model by means of *SWARM* – one of the most popular (noncommercial) applied packages for constructing the agent-oriented models are described.

Розглянуто напрями створення агент-орієнтованої моделі, яка складається з динамічно взаємодіючих трьох типів учасників електронного навчання: викладачів–авторів дистанційних курсів, тьюторів і студентів. Висвітлено підходи до створення названої моделі засобами *SWARM* – одного з найбільш популярних (некомерційних) прикладних пакетів для побудови агент-орієнтованих моделей.

Введение. Подходы к моделированию взаимодействия двух агентов, основанные на предположениях теории социальных систем Н. Лумана рассмотрены в работе [1]. Освещены направления математической формализации взаимодействия участников дистанционных курсов (ДК), размещенные в Веб-центре Львовской коммерческой академии (ЛКА) [2]. Предполагается, что равновесие на рынке электронного обучения (*e*-обучения) достигается при условиях его перехода к стационарному (или устойчивому) процессу. Цель статьи – рассмотреть подходы к разработке агент-ориентированной модели (*agent-based model*, далее – АОМ), состоящей из трех типов агентов *e*-обучения: преподавателей, тьюторов и студентов. Виртуальной средой их взаимодействия является Веб-центр ЛКА.

Агент-ориентированные модели – новое средство для добывания знаний. Основная идея АОМ – это построение «вычислительного инструмента», который представляет собой совокупность агентов с определенным набором свойств и позволяет проводить машинную имитацию реальных явлений. Доминирующим, как правило, методологическим подходом является метод, в котором вычисляется равновесие или псевдоравновесие системы, содержащей множество агентов. В то же время подобные модели, используя простые правила поведения, могут выдавать достаточно практичные результаты.

Отметим и такое направление прикладных экономических исследований, как агент-ориентированная вычислительная экономика (*Agent-Based Computational Economics, ACE*), основанная на компьютерном моделировании виртуального мира, «населенного» автономными агентами. Управление созданным согласно методологии *ACE* виртуальным миром осуществляется без вмешательства извне, т.е. лишь на основе взаимодействия агентов, владеющих способностью к обучению [3].

Таким образом, АОМ являются специализированным классом моделей, основанных на индивидуальном поведении агентов и создаваемых с целью проведения компьютерных экспериментов. Считается, что АОМ дополняют классические аналитические методы. Последние позволяют охарактеризовать равновесие исследуемой системы, а АОМ – анализировать возможность образования такого состояния.

В настоящее время в Интернете имеются специализированные издания, тематика которых непосредственно связана с разработкой АОМ. Например, международный онлайн-журнал *JASSS – Journal of Artificial Societies and Social Simulation* [4] или российский Интернет-журнал «Искусственные общества» [5].

Известно также достаточно большое количество программных продуктов, используемых для построения агент-ориентированных моде-

лей. Среди них *SWARM* – один из наиболее популярных прикладных пакетов, коллекция программных библиотек, написанных на *Objective C* группой исследователей Института Санта Фе (*Santa Fe Institute*) [6]. Некоторые из этих библиотек написаны языком скриптов, что позволяет использовать такие ключевые графические средства, как графика, окна и др. Отметим, что программы и подробная инструкция по их установке распространяются бесплатно, как *Open Source*-проект, и в свободном доступе выложены на сайте [7].

Анализ известных примеров агент-ориентированных моделей позволяет сделать следующие выводы.

- Агенты – это теоретическая (научная) абстракция, используемая в АОМ с целью представления действующих лиц, членов социально-экономической системы, принятия решений которыми имеет определенную самостоятельность. Агент может «жить» во времени, иметь графический образ, реагировать на события, поступающие от датчиков, внешних программ или пользователя (наблюдателя).

- Подавляющее большинство АОМ являются абстрактными и основная цель их создания – научный интерес, т.е. они используют условные данные и создаются для испытания нового инструмента.

- Среди моделей, рассматриваемых реальные явления, только небольшая часть имеет отношение к взаимодействиям агентов в сфере образовательных услуг.

Конечной целью создания АОМ является проведение компьютерных имитаций с целью выявления влияния флуктуаций агентов, которые действуют на микроуровне, на показатели макроуровня (в нашем случае – это совокупность определенных макроагентов национальной системы дистанционного обучения, каждый из этих агентов существенно влияет на образовательные услуги высшей школы). Основное внимание направлено на освещение подходов к построению модели взаимодействия агентов *e*-обучения на основе технологии *SWARM*.

Подходы к построению модели взаимодействия агентов *e*-обучения на основе технологии *SWARM*

Спецификация задачи построения АОМ предусматривает создание виртуального мира, в котором описывается взаимодействие трех типов агентов: *A1* – преподавателей–авторов ДК, *A2* – тьюторов, сопровождающих учебный процесс в виртуальной среде вуза и *A3* – студентов, которые являются участниками ДК. При этом используются базовые положения, определяющие взаимодействия агентов:

- взаимодействия агентов основываются на обмене информацией между ними и согласовании содержания информационных потоков с текущим состоянием системы вуза;

- агенты на основе анализа информационных взаимодействий могут выполнить такие действия:

- начать обмен информацией с определенным агентом в целях уточнения модели их совместной деятельности;

- принять сформированную модель организации их совместной деятельности как лучшую на данный момент в их области выбора;

- принять другую модель совместной деятельности и таким образом отклонить предложенную;

- агенты применяют разную институциональную поддержку и соответствующие механизмы координации своей деятельности в зависимости от степени их удаленности друг от друга в информационном пространстве системы.

Таким образом, основой взаимодействия между агентами является информация, знания, сведения, опыт. Их взаимодействие изображено на рис. 1.

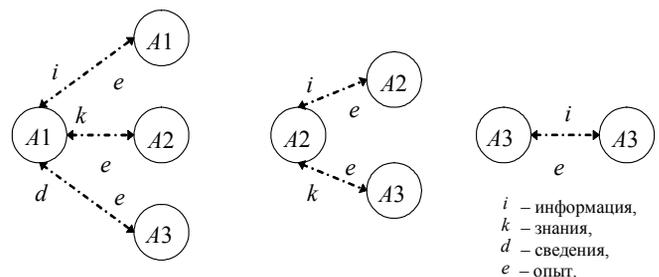


Рис. 1. Схема взаимодействия агентов *e*-обучения

В контексте технологии *SWARM* для построения АОМ и проведения на ее основе компьютерных имитаций необходимо следующее.

- Создание искусственного (виртуального) мира, включающего пространство, время и объекты, которые могут быть расположены в некоторых «точках» пространственно-временной структуры. Необходимо, чтобы эти объекты могли определять их собственное поведение в соответствии с их собственными правилами и внутренним состоянием.

- Формирование определенного количества объектов, которые будут наблюдать, записывать, анализировать данные, порождаемые поведением объектов в искусственном мире, созданном на предыдущем этапе.

- Запуск анализируемого мироздания, управляемого моделируемыми и наблюдаемыми объектами во времени по заданной определенным образом процедуре согласования.

- Взаимодействие с экспериментом АОМ через данные, которые создаются инструментальными объектами, для реализации серии контрольных экспериментальных прогонок агент-ориентированной модели.

Структура *SWARM* имеет два уровня. Первый – уровень модели, где можно строить серию моделей, вложенных друг в друга. Второй – уровень наблюдателя, рассматривающего модель (или семейство вложенных моделей) как уникальный объект для эксперимента в целях получения результатов, их визуализации, интерпретации и использования.

В соответствии с указанной технологией в типовом *Swarm*-приложении создаем высокоуровневый объект *Swarm*, который управляет заданиями по созданию, запуску и взаимодействию с нашими моделями. Объект *Swarm* не является частью мира агентов, это скорее объект нашего мира. Объект *Swarm* включает модель агента и ее мир, создающий ее определенного рода объектом, с которым можно взаимодействовать, посылая ему сообщения. Списки являются объектами и определяются *Swarm* классом *Activity*. Существует обычная процедура для создания любых *Swarm*-объектов, не зависящих от специфики модели.

На рис. 2 представлена схема создания *Swarm*-объектов для взаимодействующих агентов электронного обучения в среде Веб-центра ЛКА.

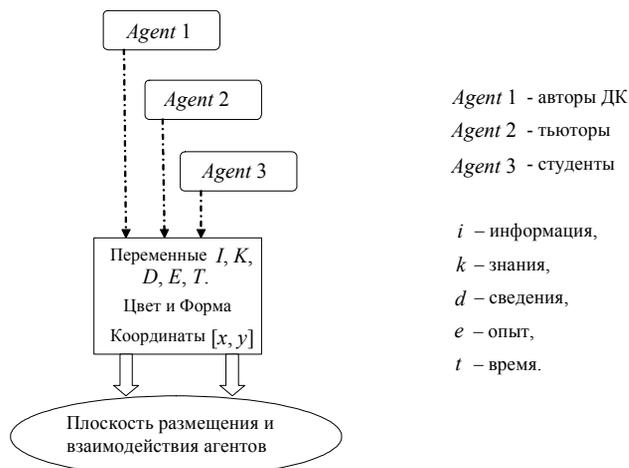


Рис. 2. Схема создания *Swarm*-объектов для агентов *e*-обучения

Объект *Swarm* обычно выполняет следующие три задания: создает разнообразные объекты, посылает объекту такие сообщения, чтобы он выполнил те действия, какие мы хотим, и объединяет все это вместе, формируя пакет, который можно запустить.

Первое задание *ModelSwarm* – создать объекты агент-ориентированной модели. Это задание выполняется при помощи метода *buildObjects()*. Заметим, что *buildObjects()* требует возвращения объекта класса *ModelSwarm*.

Второе задание (*buildActions()*) – создать группу сообщений как пакетизированный список действий и послать агенту для выполнения. Для этого списка сообщений на основе определенных подклассов класса *Activity* создается расписание, благодаря которому сообщения будут отосланы агенту в нужное время.

Последнее задание – соединение двух частей вместе для создания на базе метода *activateIn()* пакета с учетом передачи представленного *Swarm* “context’a”, в котором будет выполняться *Activity*.

Файл *ModelSwarm.java* заканчивается методом *checkTime()*. Во-первых, этот метод периодически будет отсылать показатели времени на консоль. Во-вторых, он остановит работу программы через определенный промежуток времени, определяемый переменной *endTime*. Ес-

ли работа программы еще не завершена, то на консоль отправляются текущие показатели времени. Переменной *endTime* придаем достаточно большое значение для того, чтобы позволить агенту полностью выполнить свою работу. Существуют и другие подходы к определению переменной времени, включая создание специального расписания.

Таким образом, при построении модели взаимодействия наших агентов конструктор *ModelSwarm* выполняет следующие ключевые задания:

- создает объекты, используемые в модели;
- создает группу сообщений, которые будут посылаться как пакетизированный список действий для требуемого выполнения агентом;
- обеспечивает переход в мир, где взаимодействуют между собой множество агентов.

На рис. 3 представлена схема реализации перехода от одного к большему количеству агентов.

Как видно из рис. 3, управлять коллекцией агентов можно после соответствующей их трансформации в объекты. *RunOneAgent* имплементирует агента, который прогуливается случайно по вымышленному двумерному цифровому массиву (или виртуальному миру). Матрица 80 на 80, по которой «прогуливается» агент, определяется в переменных класса *worldXSize* и *worldYSize*.

Позиция в пространстве определяется переменными *xPos* и *yPos*. Агент проводит множество случайных прогулок, в каждой из них агент сначала двигается в направлении по оси *X*, вызывая метод *randomMove()*, а затем по осе *Y*, вызывая данный метод еще раз. Метод *randomMove()* использует простой числовой генератор из математической библиотеки *Java*. Он определяет значение -1 для движения назад, 0 – для стояния на месте или $+1$ – ход вперед (*Swarm* позволяет использовать и другие генераторы случайных чисел).

Наш *OneAgents* способен выполнять два действия: прогуливаться по миру размером *X, Y* и отчитываться о месте расположения на нем в консоли. В первом методе, *randomWalk()*, агент прогуливается случайно назад или вперед по

оси *X*, а впоследствии – соответственно случайно назад или вперед по оси *Y*. В каждом случае направление передвижения определяется числом -1 , 0 или $+1$, а модульный оператор используется для того, чтобы проверить не вышел ли агент за пределы мира.

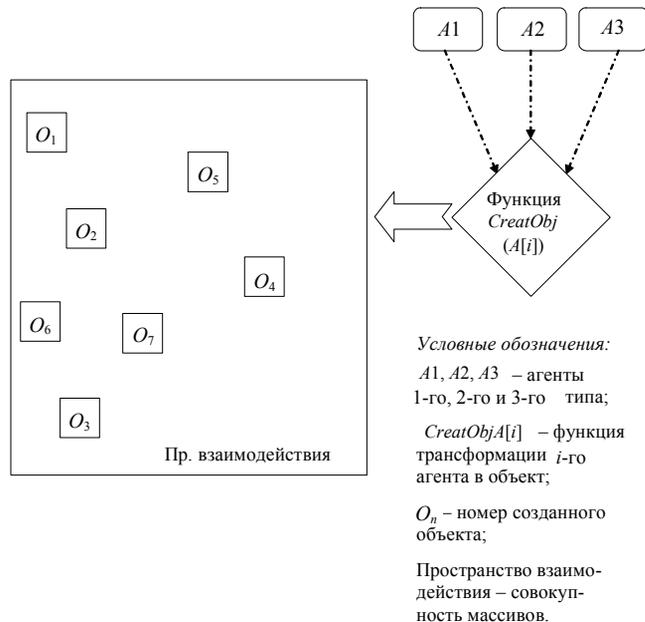


Рис. 3. Схема трансформации агентов в объекты

Swarm предоставляет большое количество инструментов для создания графического интерфейса пользователя, что позволяет визуализировать агентов в их виртуальном мире и проводить с моделью компьютерные эксперименты. Обе возможности предоставляются интерфейсом *Swarm GUISwarm*, имеющим собственную встроенную панель управления *GUI* с пятью кнопками: *START*, *STOP*, *NEXT*, *SAVE* и *QUIT*. Эти кнопки контролируют процесс имитации и нажимаются при помощи кликанья пользователем на них мышкой. Кроме того, кнопки могут быть нажаты программно.

Как только пользователь дает команду «СТАРТ», *ObserverSwarm* вызывает *model-Swarm* для построения его объектов на основе *model-Swarm.buildObjects()*. После этого *ObserverSwarm* приступает к построению собственных объектов отображения. Первым из них является объект *Colormap*, содержащий определенное пользователем соответствие между названиями цветов и числовым кодом. Объект *Colormap* соз-

дан с помощью классов выполнения *Colormap*, а коды устанавливаются с помощью метода *setColor\$ToName()*.

Содержательная интерпретация результатов моделирования

Для пересмотра значений ключевых переменных созданной агент-ориентированной модели участников *e*-обучения можно использовать системную панель управления, что позволит визуализировать агентов в их виртуальном мире и сотрудничать с моделью через эту панель управления.

На скриншоте созданной модели, представленному на рис. 4, хорошо видно, что агенты формируют свои подгруппы в центре моделируемого пространства. Взаимосвязи на конечном этапе характеризуют псевдоравновесие и возникают как результат обмена сообщениями между агентами и приобретения ими опыта.

Следовательно, содержательная интерпретация результатов АОМ указывает, что наиболее важными в Веб-центре ЛКА считаются связи между участниками *e*-обучения (авторами ДК, тьюторами и студентами). Чем более тесны эти взаимосвязи (в модельном понимании – меньшее расстояние), тем эффективнее функционирование Веб-центра ЛКА.

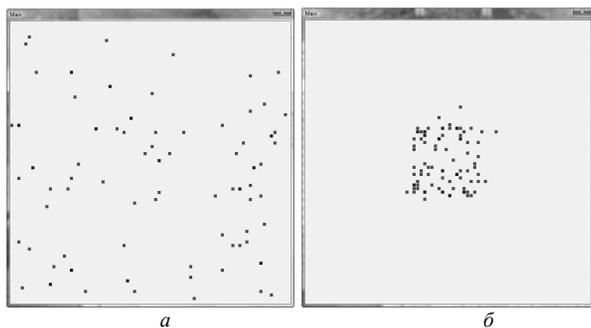


Рис. 4. Главное окно отображения результатов АОМ: а – начальный этап; б – конечный этап

Во время анализа результатов имитации уровень характеристик агентов третьего типа отличался. Один из агентов имел высокий уровень опыта, но низкий уровень знаний, другой низкий уровень опыта, однако высокий уровень знаний. Различия наблюдались и в других подгруппах, однако они были значительно меньше.

Анализируемые характеристики были предопределены наборами представленных на рис. 5 показателей, которые устанавливались для каждого типа агентов.

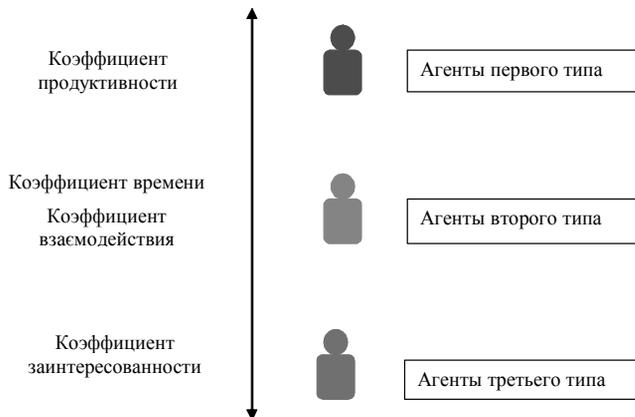
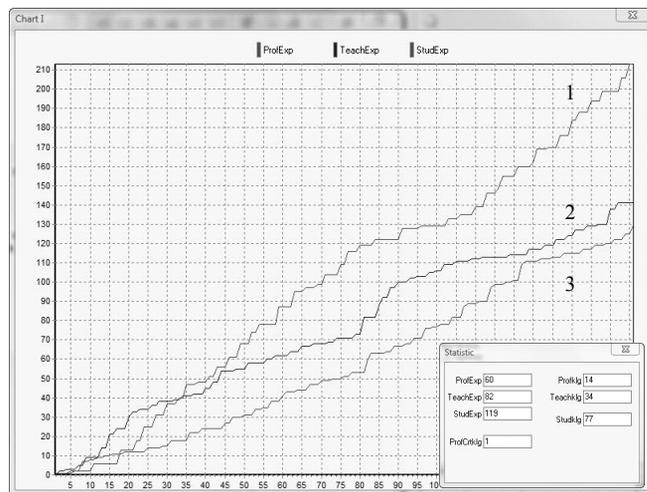


Рис. 5. Показатели взаимодействия участников *e*-обучения

Результаты компьютерной имитации взаимодействия участников *e*-обучения отображаются в окне «*Statistic*» и на соответствующих графиках (рис. 6).

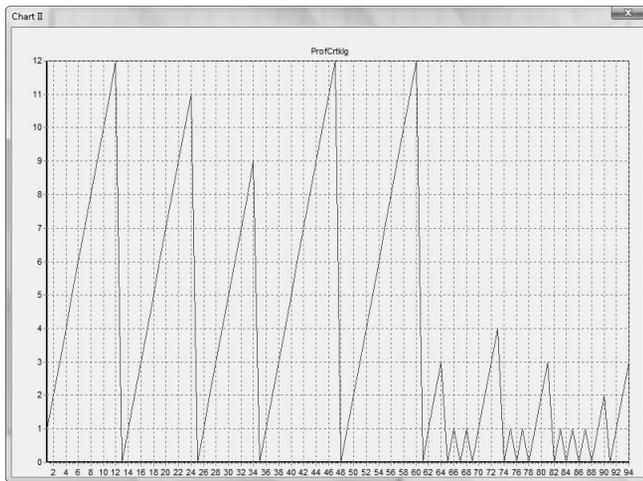
На рис. 7 представлена диаграмма продуцирования новых знаний, что присуще, в первую очередь, агентам первого типа (авторам ДК).



Условные обозначения:

- 1 – шкала представления студентов;
- 2 – шкала представления тьюторов;
- 3 – шкала представления авторов ДК;
- Ось X – время пребывания в Веб-центре ЛКА;
- Ось Y – величина суммарного опыта.

Рис. 6. Диаграмма совокупного опыта для каждой группы агентов



Условные обозначения:

- – шкала представления агентов первого типа;
- Ось X – время пребывания в Веб-центре ЛКА;
- Ось Y – количество продуцируемых знаний.

Рис. 7. Диаграмма продуцирования знаний агентами первого типа

Из рис. 7 видно, что новых знаний на первоначальной стадии *e*-обучения создавалось больше, хотя реже. В дальнейшем при формировании более близких связей процесс продуцирования новых знаний все более учащался, общий показатель уменьшался и затем приостановился на устойчивом уровне. Этот результат указывает на то, что коэффициент заинтересованности авторов ДК связан в первую очередь с их деятельностью в первой группе агентов.

Заключение. Таким образом, изложенные подходы к созданию АОМ участников *e*-обучения с помощью технологии *SWARM* позволяют сделать следующие выводы и обобщения.

- Агент-ориентированную модель участников *e*-обучения, построенную на основе технологии *SWARM*, можно рассматривать как инструмент, обеспечивающий на микроуровне любого вуза проведение исследований в сфере внедрения дистанционных образовательных технологий.

- Содержательная интерпретация результатов имитации АОМ позволяет:

- на основе визуального анализа взаимодействия агентов, участников *e*-обучения, делать выводы относительно достижения уровня равновесия (псевдоравновесия) в виртуальной учебной среде вуза;

- разрабатывать внутренние нормы и правила взаимодействия трех агентов на базе таких характеристик, как производительность и заинтересованность агентов.

- Визуальный анализ результатов моделирования показывает, что наиболее важными в Веб-центре ЛКА являются связи между участниками *e*-обучения (авторами ДК, тьюторами и студентами). Чем более тесны эти взаимосвязи (в модельном понимании – меньшее расстояние), тем эффективнее функционирование Веб-центра ЛКА.

- В дальнейшем планируем трансформировать разработанную модель в агент-ориентированную модель, которая могла бы учитывать ключевые характеристики взаимодействий анализируемых агентов на макроуровне, т.е. в контексте виртуального сетевого содружества, которое самоорганизовывается.

1. *Артеменко В.Б.* Моделювання взаємодії учасників дистанційної освіти на засадах агент-орієнтованого підходу // Вісн. Львів. комерц. акад. – Сер. екон., вип. 25 – 2007. – С. 193–197.
2. *Сайт* Веб-центра Львовской коммерческой академии. – <http://virt.lac.lviv.ua/>
3. *Site* Agent-Based Computational Economics (ACE). – <http://www.econ.iastate.edu/tesfatsi/ace.htm>
4. *Journal of Artificial Societies and Social Simulation.* – <http://jasss.soc.surrey.ac.uk/JASSS.html>
5. Журнал «Искусственные общества». – <http://www.artsoc.ru/>
6. *Site* Santa Fe Institute. – <http://www.santafe.edu/>
7. Сайт доступа к пакету *SWARM*. – <http://wiki.swarm.org/>

E-mail: artem@lac.lviv.ua
© В.Б. Артеменко, 2010