

УДК 004.4:004.89/004.94  
ГРНТИ 28.23.29

А.О. Анохин, А.В. Катаев  
Волгоградский Государственный Технический Университет

### КОНЕЧНО-АВТОМАТНАЯ МОДЕЛЬ УПРАВЛЕНИЯ ПОВЕДЕНИЕМ ИНТЕЛЛЕКТУАЛЬНЫХ АГЕНТОВ В ОБУЧАЮЩИХ ИГРАХ\*

*В работе рассматривается применение конечно-автоматной модели для моделирования поведения интеллектуальных агентов. Проводится обоснование выбора конечно-автоматной модели поведения. Описывается реализация программной платформы для моделирования поведения.*

**Ключевые слова:** игровой искусственный интеллект, интеллектуальный агент, модель поведения, обучающие компьютерные игры.

А.О. Anokhin, A.V. Kataev  
Volgograd State Technical University

### FINITE-AUTOMATON MODEL FOR CONTROLLING THE BEHAVIOR OF INTELLIGENT AGENTS IN EDUCATIONAL GAMES

*The paper considers the use of a finite-automaton model for modeling intelligent agents' behavior for educational computer games. The reasons of the use of the finite-automaton model of behavior are given.*

*The implementation of the model in a software platform is described.*

**Key words:** game AI, intellectual agent, behavior model, educational computer games

#### Введение

Моделирование поведения интеллектуальных агентов имеет большое значение в индустрии компьютерных игр. Перед разработчиком стоит непростая задача, состоящая в необходимости создания реалистичного игрового искусственного интеллекта, который будет способен демонстрировать достаточно сложное поведение персонажей, задача взаимодействия с которыми не будет для игрока тривиальной, и будет дополнять эффект присутствия. В некоторых случаях поведение персонажей копируется с реальных живых существ.

Такая задача стоит перед разработчиками компьютерных игр с тех пор, как игры стали популярными. В зависимости от жанра игры агенты могли быть наделены различными способностями. Так, поведение неигровых персонажей игры Doom сводится к простому правилу: атаковать игрока при его обнаружении. Поведение агентов в обширных ролевых играх, как, например, Morrowind или Gothic, гораздо сложнее и учитывает различные факторы, такие как текущее время суток, собственное расписание действий агента, его цели и взаимоотношения с другими агентами и игроком, а также многие другие факторы.

За период с 2014 по 2019 год в различных наукометрических базах можно найти более 50 тысяч работ тематике моделирования поведения интеллектуальных агентов. В некоторых работах ведётся рассмотрение потенциальных возможностей агентного подхода в разработке видеоигр [1] или способностей агентов решать конкретные проблемы, как в работе [2], где моделируется поведение агентов, направленное на эффективное управление ресурсами. В некоторых работах рассматриваются конкретные подходы к моделированию поведения: так, в источнике [3] описываются нюансы работы с деревьями поведения, а в работе [4] проводится создание автономного агента для прохождения игры Super Mario Bros. на базе конечных автоматов.

В [5] агентный подход применяется для моделирования поведения толпы. Выполнение этой задачи может иметь исключительную важность, например, для случая, когда нужно спроектировать эвакуационные выходы крупного торгового центра, и необходимо получить представление о том, как будет двигаться большое количество людей. Также задача моделирования поведения агентов важна для робототехники, и авторы работы [6] описывают построение виртуального полигона для отработки взаимодействия группы роботов.

В настоящее время появились исследования, в которых рассматривается применение интеллектуальных агентов при разработке так называемых «серьёзных» игр (*serious games*) [7-9], в част-

\* Работа выполнена при финансовой поддержке РФФИ в рамках научного проекта № 18-07-01308а

ности, обучающих игр [10]. В работе [11] рассматривается способ построения среды разработки интеллектуальных агентов, которых впоследствии можно использовать для различных задач. Для обучающих компьютерных игр, где неигровые персонажи (non-player character, NPC) используются как основной способ передачи знаний игроку и получения им тестовых заданий [12,13], а порядок изучения материала может быть нелинейным, требуется подчинить поведение персонажа обучающему курсу, описываемому строгой математической моделью [14,15].

Работа посвящена применению агентного моделирования для задачи создания натуралистичного поведения неигровых персонажей в играх.

### Выбор и обоснование выбора способа моделирования поведения

К настоящему моменту существует несколько основных способов реализации модели поведения интеллектуальных агентов [17, 18].

Исторически, первым появился способ реализации модели поведения на основе правил [19]. Агент имел чёткие установки, которые определяли его поведение в каждой конкретной ситуации. Такой подход был прост в реализации, однако имел существенный недостаток: поведение агента становилось детерминированным, в каждой повторяющейся ситуации он повторял собственные действия, что позволяло предугадывать поведение агента наперёд. Классическим примером такой реализации является игра Pac-Man, разработчики которой должны были реализовать сложное поведение в условиях ограниченных вычислительных возможностей. Поведение каждого из приведений, преследующих персонаж игрока, основано на наборе простых правил, несколько отличающихся для разных приведений. В результате, из четырёх приведений, только одно следовало за ним, остальные приведения передвигались по игровому миру сложным, но предсказуемым образом. Иллюстрация этого поведения представлена на рисунке 1.

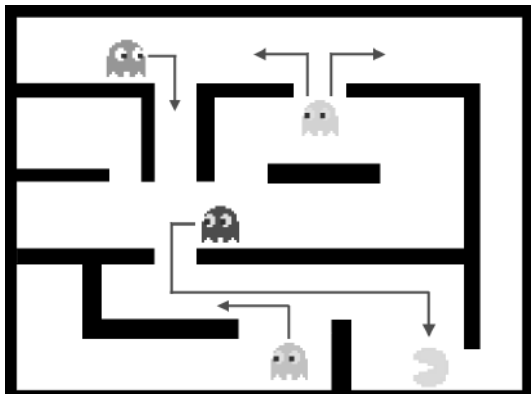


Рис. 1. Иллюстрация поведения на основе правил

необходимо поесть. Первоначально происходит проверка, есть ли у агента с собой пища, после чего, в зависимости от того, есть она или рюкзак пуст, агент либо съедает запас пищи из рюкзака, либо отправляется на поиски пищи. Процесс поиска пищи можно декомпозировать на непосредственно поиск и действие по подъёму пищи (или, например, снятия плодов с дерева или ягод с куста). Процесс нахождения пищи также можно декомпозировать, таким образом, дерево будет более глубоким и ветвистым.

Более сложное поведение обычно описывается с помощью конечных автоматов. Одна из современных вариаций этого подхода носит название GOAP (Goal-Oriented Action Planning) и представляет собой частный случай конечного автомата, когда в каждом состоянии агент формирует цель, которой нужно достигнуть, после чего строит последовательность действий, направленную на достижение этой цели. Такой подход был впервые применён в компьютерной игре F.E.A.R., вышедшей в 2005 году. Такой подход впоследствии был применён и в других успешных игровых проектах, таких как S.T.A.L.K.E.R., Fallout 3 и Deus Ex: Human Revolution.

Другим подходом к реализации модели поведения является построение деревьев поведения. Деревья поведения – это ориентированный ациклический граф, узлами которого являются возможные варианты поведения агента. Ширина дерева характеризует количество доступных действий, а длина ветвей (глубина дерева) – на их сложность. Деревья поведения имеют существенное преимущество перед конечными автоматами, которое заключается в том, что с ростом числа доступных состояний сложность дерева растёт не так быстро, как сложность автомата.

Вариант применения деревьев поведения представлен на рисунке 2. На этом рисунке приведён пример дерева поведения для случая, когда агенту

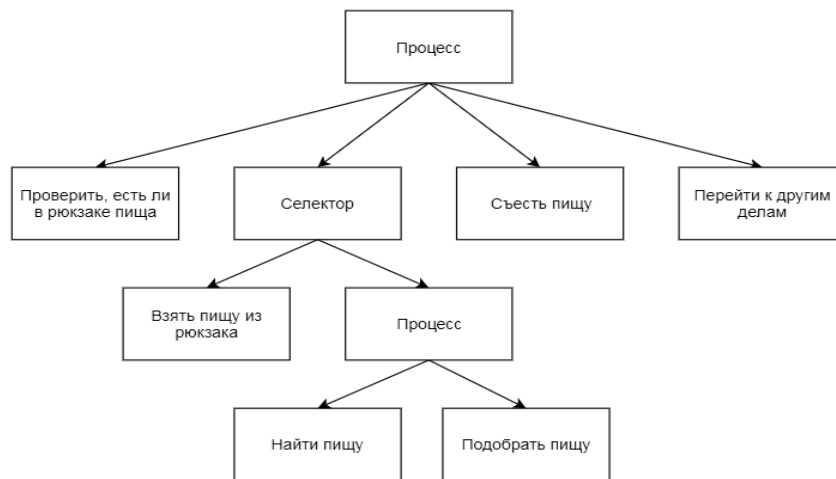


Рис. 2. Пример дерева поведения

### Описание реализации программной платформы

Для реализации модели поведения было принято решение использовать язык программирования C++, мультимедийную библиотеку SFML и библиотеку TGUI для создания графического интерфейса пользователя.

Перед построением модели поведения были проведены работы над самой платформой. Был запрограммирован динамический игровой мир, условия в котором могли изменяться и оказывать влияние на поведение агентов. В числе таких изменяющихся факторов смена дня и ночи, смена сезона и погоды, а также положение игровых объектов.

Следующим шагом стало проектирование модели поведения. Выбранный подход GOAP предполагает наличие у агентов списка состояний, в которых они могут находиться, списка целей, которые нужно достигать в каждом конкретном состоянии, списка действий, которые служат для достижения цели, а также списка факторов, которые диктуют агенту такие цели.

Список состояний определялся целями, которых нужно было достичь в этих состояниях. Список целей состоял из следующих элементов: бездействовать (Idle); спать (Sleep); есть (Eat); пить (Drink); прятаться (Hide); убегать (Flee); отдыхать (Rest).

Каждой цели был присвоен свой приоритет. Самый низкий приоритет получила цель «бездействовать». Наиболее высокий приоритет получила цель «убегать» в значении спасения собственной жизни от агрессора. Далее, в порядке убывания, приоритеты были следующими: «прятаться», «пить», «есть», «спать», «отдыхать». Таким образом, чем более важным для выживания существа в реальной жизни является действие, тем больший приоритет оно получило.

Начальным состоянием агента является состояние Idle (бездействие). В этом состоянии агент находится, пока не срабатывают какие-либо факторы, которые продиктуют ему новую цель и переведут в другое состояние. Получив цель, приоритет которой больше, чем у текущей, агент формирует последовательность действий, направленную на достижение цели. Когда агент достигнет цели, произойдёт проверка, нужно ли выполнять какую-либо другую цель, приоритет которой ниже, или же можно ничего не делать – в таком случае агент вернётся в состояние Idle и будет находиться в нём, пока не сработает новое условие.

Визуальное представление полученной модели поведения приведено на рисунке 3. На рисунке приведены возможные состояния агента и их приоритет. Модель построена таким образом, что в случае поступления сигнала на переход в состояние с более высоким приоритетом агент перейдёт в него моментально, но в случае поступления сигнала с более низким приоритетом агент сначала перейдёт в состояние бездействия, после чего займёт требуемое состояние.

Процесс построения последовательности действий зависит от текущего состояния агента, а также факторов, определяющих его состояние. В числе таких факторов могут быть определённые условия. Например, цель агента состоит в необходимости поесть. В таком случае происходит про-

верка условия, есть ли у агента запас еды, найденный ранее. Если есть, то можно не искать пищу, а съесть имеющуюся. Если нет, то необходимо найти еду и съесть её.

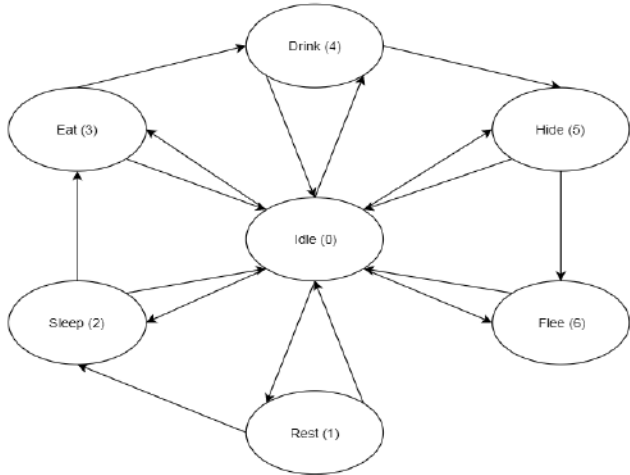


Рис. 3. Конечно-автоматная модель поведения (в скобках указаны приоритеты состояний)

щего состояния агента, которое обуславливает первое действие в цепочке, до последнего, которое обусловит достижение цели.

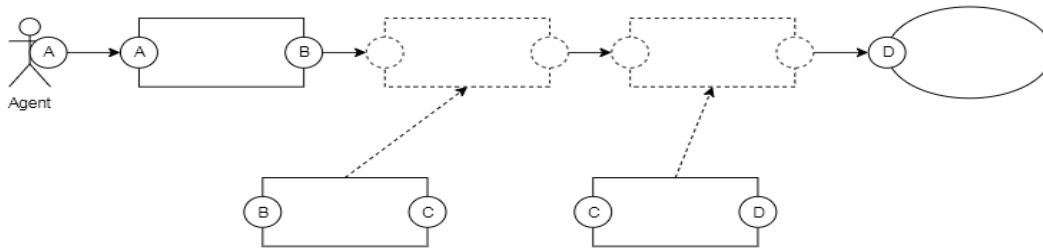


Рис. 4. Построение цепочки действий

В результате проектирования была получена модель поведения на основе конечного автомата. Модель была внедрена и протестирована на базе программной платформы, созданной заранее.



Рис. 5. Скриншот выполнения программы

В разрабатываемой игре взаимодействие персонажа игрока с неигровыми персонажами является основным способом получения теоретических материалов и выполнения обучающих игровых

Визуально процесс построения цепочки действий представлен на рисунке 4. Действия представлены прямоугольниками, каждое действие имеет предусловие и постусловие, изображённые кружками в левой и правой частях действия. Действия могут быть сформированы в последовательность только в том случае, если постусловие предыдущего действия совпадает с предусловием следующего. Выполнение цели, изображённой эллипсом, обуславливается достижением условия, обозначенного кружком в левой части эллипса. Таким образом, агенту необходимо сформировать цепочку действий, последовательно проверяя предусловия и постусловия каждого действия от текущего состояния агента, которое обуславливает первое действие в цепочке, до последнего, которое обусловит достижение цели.

Скриншоты выполнения программы представлен на рисунке 5. Агенты представлены цветными прямоугольниками с эмоциями, в правой части экрана есть список всех агентов и их состояний и состояния мира к настоящему моменту.

### Заключение

В рамках исследования был произведён обзор основных подходов к моделированию поведения интеллектуальных агентов, была выбрана и реализована конечно-автоматная модель для описания поведения неигровых персонажей.

Разработанная модель и программное обеспечение будет использоваться при создании обучающей компьютерной игры.

заданий. Использование приоритетов при задании состояний неигровых персонажей позволит гарантировать, что они адекватно отреагируют на взаимодействие с персонажем игрока.

### **Литература**

- 1 Pacheco-Morales A. An Agent-Based Model for Game Development //Technology Trends. 2018. С. 130.
- 2 Cascalho J., Mabunda P. Agent-Based Modelling for a Resource Management Problem in a Role-Playing Game //Portuguese Conference on Artificial Intelligence. – Springer, Cham, 2015. С. 696–701.
- 3 Marcotte R., Hamilton H.J. Behavior trees for modelling artificial intelligence in games: A tutorial // The Computer Games Journal. 2017. Т. 6. № 3. С. 171–184.
- 4 Mora A.M. et al. Creating autonomous agents for playing Super Mario Bros game by means of evolutionary finite state machines // Evolutionary Intelligence. 2014. Т. 6. № 4. С. 205–218.
- 5 Бекларян А.Л., Акопов А.С. Моделирование поведения толпы на основе интеллектуальной динамики взаимодействующих агентов // Бизнес-информатика. 2015. № 1 (31).
- 6 Галин Р.Р., Кулагин К.А. Использование виртуального полигона для отработки взаимодействия роботов в многоагентной робототехнической системе // Модели мышления и интеграция информационно-управляющих систем (ММИИУС-2018). 2018. С. 208–212.
- 7 Di Ferdinando A. et al. Agent based modelling to build serious games: The learn to lead game //International Work-Conference on the Interplay Between Natural and Artificial Computation. – Springer, Cham, 2015. С. 349–358.
- 8 Gentile M. et al. An Agent Based Methodology to Design Serious Game in Social Field //International Conference on Games and Learning Alliance. – Springer, Cham, 2013. С. 145–156.
- 9 Blatsios S., Refanidis I. Towards an Adaption and Personalisation Solution Based on Multi Agent System Applied on Serious Games // IFIP International Conference on Artificial Intelligence Applications and Innovations. – Springer, Cham, 2019. С. 584–594.
- 10 Tsatsou D., Vretos N., Daras P. Adaptive game-based learning in multi-agent educational settings // Journal of Computers in Education. 2019. Т. 6. № 2. С. 215–239.
- 11 Goel A.K., Rugaber S. Interactive meta-reasoning: Towards a CAD-like environment for designing game-playing agents // Computational creativity research: Towards creative machines. – Atlantis Press, Paris, 2015. С. 347–370.
- 12 Шабалина О.А. Применение 3i-подхода для разработки обучающих игр по объектно-ориентированному программированию / О.А. Шабалина, П.Н. Воробкалов, А.В. Катаев // Вестник компьютерных и информационных технологий. 2011. № 6. С. 46–52 + 3-я стр. обл.
- 13 Shabalina O. 3I-Approach for IT Educational Games Development / Shabalina O., Vorobkalov P., Kataev A., Tarasenko A. // Proceedings of the 3rd European Conference on Games-Based Learning, Graz, Austria, 12–13 October 2009 / FH JOANNEUM University of Applied Science. – [UK], 2009. P. 339–344.
- 14 Shabalina O. Development of Educational Computer Games: Learning Process Model and how it is Integrated into the Game Context / O. Shabalina, P. Vorobkalov // World Applied Sciences Journal 24 (Information Technologies in Modern Industry, Education & Society). – IDOSI Publications, 2013. P. 256–267.
- 15 Шабалина О.А. Разработка обучающих компьютерных игр: как сохранить баланс между обучающей и игровой компонентой? / О.А Шабалина // Образовательные технологии и общество. 2013. № 16 (3). С. 586–602.
- 16 Merrick K.E. Game-playing agents and non-player characters //Computational Models of Motivation for Game-Playing Agents. – Springer, Cham, 2016. С. 45–65.
- 17 Jeffery C., Al-Gharaibeh J. Non-Player Characters and Quests // Writing Virtual Environments for Software Visualization. – Springer, New York, NY, 2015. С. 127–138.
- 18 Tsalgatidou A., Loucopoulos P. Rule-based behaviour modelling: specification and validation of information systems dynamics //Information and Software Technology. 1991. Т. 33. № 6. С. 425–432.

**Сведения об авторах**

**Александр Олегович Анохин**  
Магистрант  
ВолГТУ  
Россия, Волгоград  
Эл. почта: alex.anokhin.st@gmail.com

**Александр Вадимович Катаев**  
канд. техн. наук, доц.  
ВолГТУ  
Россия, Волгоград  
Эл. почта: alexander.kataev@gmail.com

**Information about authors**

**Alexander Anokhin**  
Master student  
Volgograd State Technical University  
Russia, Volgograd  
E-mail: alex.anokhin.st@gmail.com

**Alexander Kataev**  
PhD, Associated professor  
Volgograd State Technical University  
Russia, Volgograd  
E-mail: alexander.kataev@gmail.com

УДК 004.056.57  
ГРНТИ 81.93.29

**Т.И. Вишневская, А.С. Иванов**  
МГТУ им. Н.Э. Баумана

**ГИБРИДНЫЙ МЕТОД ПРИНЯТИЯ РЕШЕНИЙ ИНТЕЛЛЕКТУАЛЬНЫМ АГЕНТОМ ПРИ ФОРМИРОВАНИИ ПУТИ В НЕИЗВЕСТНОМ ОКРУЖЕНИИ**

*В работе представлен вариант решения задачи навигации в неизвестном окружении с помощью использования нейро-нечеткой сети типа ANFIS. Описаны основные методы решения задачи, структура нейро-нечеткой сети, обучающая выборка, проблемы, возникшие во время обучения модели. Продемонстрирован результат работы обученной модели в тестовом окружении. Описаны возможные варианты применения разработанного метода.*

**Ключевые слова:** навигация, неизвестное окружение, нейронные сети, нечеткая логика, ANFIS.

**T.I. Vishnevskaya, A.S. Ivanov**

Bauman Moscow State Technical University

**HYBRID DECISION MAKING METHOD FOR PATHFINDING IN UNKNOWN ENVIRONMENT**

*The paper presents a method for solving the navigation problem in an unknown environment by using ANFIS neuro-fuzzy network architecture. Paper discusses methods for solving described problem, the structure of the neuro-fuzzy network, training data and problems that arose during the training of the algorithm. The resulting model is tested in simulation environment. Possible applications of the developed method are described.*

**Keywords:** navigation, unknown environment, neural networks, fuzzy logic, ANFIS.

**Введение**

Задача поиска пути возникает в самых различных областях – при непосредственно навигации на местности, например, при прокладывании маршрутов; при маршрутизации и организации передачи данных в сети; при проектировании печатных плат. В некоторых случаях полученный результат может не являться путём в привычном смысле слова, а представлять собой лишь некоторый набор элементов, связанных определенным отношением. Задача поиска пути в неизвестном окружении возникает при разработке программного обеспечения для беспилотных автомобилей и автономных роботов.

Существуют различные группы методов решения задачи навигации в неизвестном пространстве, например, методы, основывающиеся на представлении пространства в виде графа, которые сводят задачу навигации к исследованию окружающей среды и решения задачи поиска пути в графе. Также используются методы обучения на основе нейронных сетей, использующие обучение с подкреплением [1], в котором нейронная сеть обучается на симуляторе, а каждое действие, принятое моделью во время обучения, поощряется функцией награды. Возможен вариант решения задачи с помощью использования нечетких систем, которые описывают входные параметры системы с