

УДК 004.896

## АНАЛИЗ ПРОБЛЕМ МОДЕЛИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНОГО ПОВЕДЕНИЯ ПЕРСОНАЖЕЙ В КОМПЬЮТЕРНЫХ ИГРАХ

*Поспелов С.М., Бондаренко И.Ю.  
Донецкий национальный технический университет*

Игровой искусственный интеллект — набор программных методик, которые используются в компьютерных играх для создания иллюзии интеллекта в поведении персонажей, управляемых компьютером [1].

Первые компьютерные игры не имели одиночного режима игры, поэтому не требовали реализации искусственного интеллекта. Но начиная уже с таких игр как Star Trek и Pac-Man, средства ИИ стали неотъемлемой частью любой компьютерной игры. За полвека развития компьютерных игр заметно вырос уровень графики и физики, но средства ИИ используются те же, что и много лет назад. Это так называемый метод конечных автоматов и сценарии. Данные методики практически зашли в тупик в своём развитии [3, стр. 48-49]. Более эффективными методами моделирования персонажей являются методы, использующие нейронные технологии, нечеткую логику и генетические алгоритмы. На данный момент эти методы только начинают внедрять в компьютерные игры и их исполнение ещё далеко от идеального, поэтому изучение и усовершенствование их является очень важной и актуальной задачей [2].

Целью данной работы является изучение и анализ методов моделирования интеллектуального поведения персонажей в компьютерных играх.

Самый первый метод, с появлением которого связывают появление ИИ в компьютерных играх — метод детерминированных конечных автоматов. Это модель поведения, составленная из конечного ряда состояний и условий, при котором персонаж переходит из одного состояния в другое [4].

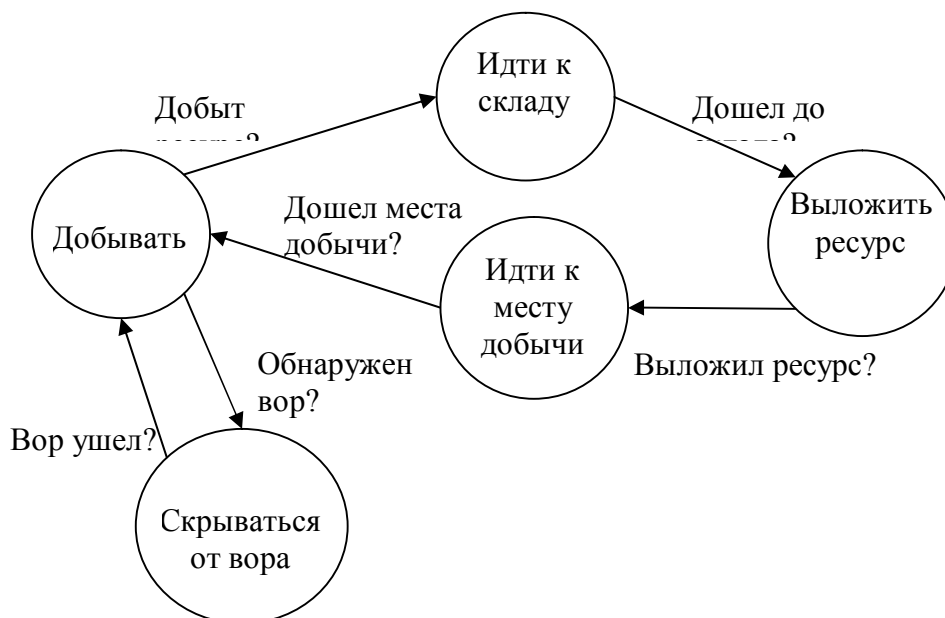


Рисунок 1 – Граф переходов детерминированного конечного автомата

Например, у нас есть такая ситуация: некоторый персонаж добывает некоторый ресурс, после того как он добыл некоторое его количество, он должен отнести его на склад. Состояниями конечного автомата будут: «Добывать», «Идти к складу»,

«Выложить ресурс», «Идти к месту добычи», «Скрываться от вора». Правилами перехода: если «Добываю» и «добыл ресурс», тогда «Идти к складу»; если «Иду к складу» и «дошел», тогда «Выложить ресурс»; если «Добываю» и «обнаружен вор», тогда «Скрываться от вора». Графически данный конечный автомат изображен на рис. 1.

Такой автомат никак не связан с традиционным пониманием искусственного интеллекта, но может быть признан реалистичным с точки зрения игрока, поскольку персонаж, управляемый конечным автоматом, выполняет ожидаемые действия. Правда, при этом не будет эффекта непредсказуемости, который наблюдается, когда персонажем управляет человек.

Для введения элемента случайности добавляются вероятности переходов в разные состояния при одном и том же условии. Например, в случае появления вора, персонаж в 70% случаях убежит, а в 30% случаях попытается его прогнать. Такой конечный автомат называется недетерминированным, его графическое представление изображено на рис. 2.

Метод конечных автоматов был использован для моделирования поведения персонажей в таких играх как Doom, Quake и Wolfenstein [2].

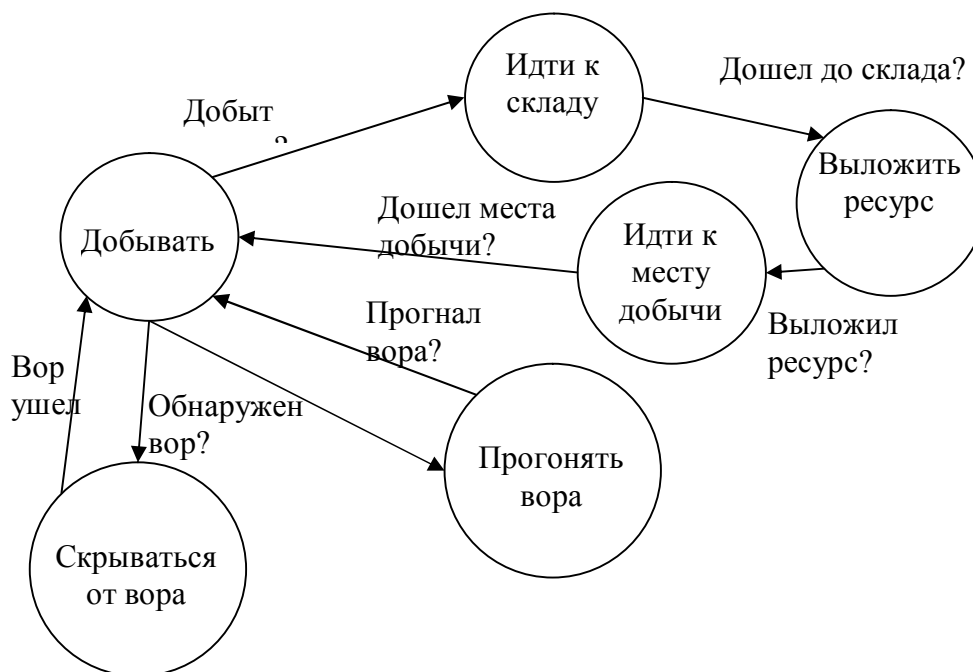


Рисунок 2 – Граф переходов недетерминированного конечного автомата

Метод конечных автоматов накладывает некоторые ограничения на возможность переходов между состояниями. Так, персонаж не может перейти одновременно в несколько состояний. В таких случаях можно применить другую технологию — систему продукционных правил. Все поведение персонажа описывается набором простых правил вида: условие->действие. При этом условие может содержать в себе несколько предикатов. Правила объединяются в деревья решений [2]. Пример дерева решений изображен на рисунке 3.

Системы правил применяются, в основном, в стратегических играх, например в Age of Empires [2].

Одним из современных подходов к моделированию поведения персонажей в компьютерных играх является использование генетического алгоритма. Он основан на

использовании естественного отбора, под воздействием которого выживают только самые пригодные индивидуумы. Для этого создается набор персонажей, имеющих разные наборы характеристик (генов). В процессе отбора из набора выбираются два родительских персонажа, показавших себя наилучшим образом. Далее создаются персонажи с генами, полученными в результате скрещивания генов родителей. Для развития эволюции некоторые гены могут мутировать, т.е. принимать случайные значения. После создания нового набора отбор повторяется. Так, после множества итераций мы получаем персонажа с поведением, близким к оптимальному [3, стр 481-508].



Рисунок 3 – Пример дерева решений

Примером использования генетических алгоритмов могут служить игры The Sims и GTA [2].

Другим примером нестандартной реализации ИИ в компьютерных играх является нечетная логика. В отличие от классической логики, где значение может быть истинным или ложным, в нечеткой логике переменные могут принимать такие значения, как почти полностью истинное или не совсем ложное. Практически это может использоваться для моделирования рассуждений человека, когда знания о каком-то предмете на совсем точные. Нечеткие системы особенно подходят для обеспечения плавного управления и принятия решений с частичными истинностными значениями [3, стр. 449-468].

Нейронные сети являются одной из самых перспективных технологий создания ИИ. Идея заключается в попытке математического моделирования работы головного мозга человека, который состоит из множества нейронов, соединенных между собой. Процесс моделирования состоит из двух основных этапов: обучение и эксплуатация. На математическом уровне обучение происходит путем подбора системой нужных весовых коэффициентов в уравнениях функционирования сети. После некоторого времени обучения нейронная сеть уже готова к использованию, а также последующему обучению.[2] Принцип работы заключается в следующем: на нейроны первого слоя поступают некоторые входные сигналы, вычисляется их взвешенная сумма, которая подвергается нелинейному преобразованию с помощью функции активации, после чего сформированные выходные значения подаются либо на входы нейронов следующего слоя, либо на выходы нейросети. Правильность работы нейронной сети зависит от весовых коэффициентов, а обучение состоит в нахождении их оптимальных значений. В компьютерных играх на вход подаются такие параметры, как, например, тип противника, тип его оружия, расстояние до противника, количество его жизней.

Выходным значением может быть: тип оружия, которое стоит использовать против данного врага, или действие, которое необходимо совершить в текущей ситуации.

Далее приведена сравнительная таблица методов моделирования поведения персонажей в компьютерных играх.

Таблица 1 – Сравнение методов моделирования поведения персонажей в играх.

Название метода	Преимущества	Недостатки
Конечные автоматы	Простота в понимании, написании и отладке небольших систем, высокая производительность в работе.	Реализация больших систем занимает очень много времени и тяжело расширяется. Невозможность нахождения в нескольких состояниях одновременно. Только имитация интеллектуальности персонажа.
Система продукционных правил	Простота понимания и расширения, гибкость, широкая область применения.	Неустойчивость к зашумлению входных данных, ограниченные возможности языков представления знаний, низкая эффективность в крупных системах, для накопления знаний необходимо привлекать экспертов.
Генетический алгоритм	Всегда находит глобальное оптимальное значение.	Во время обучения возможна медленная сходимость, нет формальных методик определения операций генетического алгоритма (скрещивания, мутации).
Нечеткая логика	Находит правильный результат в условиях НЕ-факторов (при зашумленных и неполных входных данных), более точная имитация человеческих рассуждений, простота разработки и расширения.	Значительные затраты вычислительных ресурсов, сложность при создании функций принадлежности, экспоненциальное увеличение правил при линейном увеличении переменных.
Нейронные сети	Находит правильный результат при зашумленных входных данных, обеспечивает хорошую аппроксимацию, способность к адаптации, широкий набор доступных алгоритмов обучения.	Обучение занимает много времени, большие затраты вычислительных ресурсов, подготовка достаточно представительного обучающего множества

Благодаря таким достоинствам, как адаптивность и устойчивость к шумам, нейронные сети представляются наиболее перспективной технологией имитации интеллектуального поведения персонажей в компьютерных играх. Дальнейшие исследования будут направлены на разработку конкретных нейроалгоритмов управления персонажами в играх аркадного типа и методик оценки интеллектуальности поведения этих персонажей.

### Литература

- [1] Игровой искусственный интеллект. // Википедия [Электронный ресурс]. – Режим доступа: [http://ru.wikipedia.org/wiki/Игровой\\_искусственный\\_интеллект](http://ru.wikipedia.org/wiki/Игровой_искусственный_интеллект). Дата обращения: 06.10.2010.
- [2] Сказ о том, как искусственный интеллект в играх делают // Спецвыпуск: Хакер – 2006 - №3 – с.38-41.
- [3] Шампадар, Алекс Дж. Искусственный интеллект в компьютерных играх: как обучить виртуальные персонажи реагировать на внешние воздействия. :Пер. с англ. – М. : ООО «И.Д. Вильямс», 2007. – 768с.
- [4] Finite state machine. // Википедия [Электронный ресурс]. – Режим доступа: [http://en.wikipedia.org/wiki/Finite-state\\_machine](http://en.wikipedia.org/wiki/Finite-state_machine). Дата обращения: 06.10.2010.