# Исследование методов моделирования принятия решений на примере ботов в компьютерной игре

# Введение

# Постановка задачи и обоснование актуальности

## Принятие решений

Принятие решений – процесс построения множества альтернативных возможностей и выбор одного или нескольких элементов из этого множества в соответствии с определенной системой ценностей и предпочтений. Этот процесс может быть автоматизирован с помощью вычислительной техники. Вычислительная система, принимающая решение и действующая в соответствии с ним, называется агентом. Если при этом агент стремится достичь наилучшего результата, такой агент называется рациональным [Рассел и Норвиг].



Рисунок 1. Модель агента.

На рисунке 1 представлена модель агента. Агент получает информацию о внешней среде и обновляет свое внутреннее состояние. Затем на основании имеющейся информации и определенной системы правил он производит выбор решения и выполняет связанное с ним воздействие на среду.

В простейшем случае внутреннее состояние может отсутствовать. Такой агент называется простым рефлексным агентом. В этом случае действия, производимые агентом, являются простыми реакциями на внешнюю среду и не зависят от прошлого состояния среды и прошлых действий агента. Такая стратегия оправдана, если датчики всегда предоставляют информацию, которой достаточно для принятия решения.

В случае частичной наблюдаемости среды агент должен поддерживать внутреннее состояние, которое зависит от истории наблюдений и отражает некоторые из ненаблюдаемых аспектов текущего состояния среды. Для корректного обновления внутреннего состояния нужно обладать дополнительной информацией о том, как среда изменяется независимо от агента, и о том, как те или иные действия агента изменяют среду. Эта информация, заложенная в агенте, называется моделью мира, а такой агент – рефлексным агентом, основанным на модели.

Процесс принятия решений можно представить в виде отображения [Шампандар]:

,

где C – контекст, S – стратегия принятия решений, D – принятое решение. Оно показывает, что в соответствии со стратегией S, реакцией агента на окружение C является действие D.

Возможно альтернативное отображение, моделирующее принятие решений:

,

где R – вещественное число, характеризующее пригодность принятого решения. Каждой комбинации контекста, стратегии и возможного решения ставится в соответствие значение пригодности решения для представленных условий. Оптимальным решением при этом является то, которое при равных C и S имеет максимальную пригодность.

## Компьютерные игры

Компьютерные игры или видеоигры – это компьютерные программы для организации игрового процесса. Назначение видеоигр схоже с назначением произведений литературы и кинематографа: они могут использоваться для развлечения, образования или передачи авторской идеи.

Игровая индустрия или индустрия интерактивных развлечений – это один активно развивающихся секторов экономики. Согласно прогнозам аналитиков, оборот игровой индустрии к 2018 году достигнет 96 миллиардов долларов [Video Game Market Overview. DFC Intelligence. Апрель 2016]. Бюджеты крупных игровых проектов сравнялись с бюджетами фильмов-блокбастеров [http://www.scotsman.com/lifestyle/gadgets-gaming/new-gta-v-release-tipped-to-rake-in-1bn-in-sales-1-3081943].

Разработка видеоигр – это технологически сложный процесс. Развитие видеоигр также способствует развитию многих компьютерных наук: компьютерная графика, моделирование физических процессов, алгоритмы и структуры данных, искусственный интеллект и другие.

## Боты в компьютерных играх

В общем случае бот – это компьютерная программа, выполняющая какие-либо действия через интерфейсы, предназначенные для людей. В частном случае компьютерных игр бот – это программа, имитирующая поведение живых игроков в многопользовательских играх. Это агент, который анализирует окружающую обстановку, принимает решения и выполняет действия, которые дают ему игровые преимущества и приводят к выигрышу в соответствии с заданными правилами игры.

Искусственный интеллект, применяемый в разработке видеоигр, является слабым. В то время, как сильный искусственный интеллект является имитацией человеческого интеллекта, формирующего решения для новых неизвестных задач, слабый является набором алгоритмов, предназначенных для решения конкретного заранее заданного набора задач. В компьютерных играх набор решаемых задач зависит от жанра игры и роли агента в этой игре.

Благодаря большой вычислительной мощности современных компьютеров боты могут принимать более эффективные решения чем игроки и с более высокой скоростью. Однако, так как основной задачей видеоигр является развлечение игрока, боты должны имитировать поведение живого человека, играющего с ним на равных. В противном случае игра против оппонента, которого практически невозможно победить, быстро приведет к потере интереса.

## Постановка задачи

В данной работе рассматривается пример игры в жанре «шутер с видом сверху» (top-down shooter). В играх этого жанра игроки управляют вооруженными персонажами, сражаются с другими игроками или ботами и наблюдают за полем боя с высоты птичьего полета. Графика может быть как двухмерной, так и трехмерной, но игровая логика как правило рассчитывается в двухмерной координатной системе.

{добавить скриншоты из разрабатываемой игры, когда она появится}

Правила игры:

* Игроки обладают ограниченной дальностью и углом обзора.
* В игре есть стены и укрытия. За стеной не видно других игроков. За укрытиями видно, но все выстрелы могут с вероятностью 50% быть заблокированными укрытием, если выстреливший игрок стоит от него дальше, чем на 1 метр.
* Игроки могут делиться на команды.
* Игра ведется до поражения всех оппонентов. Выигрывает команда, в которой остались игроки.

В терминах теории игр рассматривая игра:

* симметричная;
* кооперативная;
* с нулевой суммой;
* непрерывная;
* параллельная;
* с неполной информацией.

Существует большое количество игр этого жанра, но во всех этих играх в качестве врагов игрока используются слабые монстры, которые нападают со всех сторон и пытаются одолеть игрока количеством. В них в качестве развлекательного фактора используется постоянное движение и стрельба по большому количеству целей. В данной работе предлагается рассмотреть симметричную многопользовательскую игру, в которой все участники, как живые, так и компьютерные, находятся в равных условиях. В этом случае победитель определяется навыками: быстротой реакции, качеством тактических решений, меткостью стрельбы.

В обычных шутерах с видом сверху враги, управляемые компьютером, должны просто идти в сторону игрока и стрелять в его направлении. Для разработки искусственного интеллекта ботов для игры с правилами, рассмотренными выше, требуется использовать более сложные методы и алгоритмы, а сами боты должны решать большее количество задач: поиск оптимального пути, поиск оптимальной тактической позиции, выбор стратегии, реализация тактических маневров, прицеливание и стрельба, мультиагентное взаимодействие и другие.

# Обзор методов и алгоритмов принятия решений, применяемых в компьютерных играх

## Рулевое управление (steering behavior)

[Reynolds, Buckland]

## Потенциальные поля / тепловые карты

## Поиск в ширину

## Алгоритм Дейкстры

{Курсовик по алгоритмам и структурам данных}

## Алгоритм A\*

{Курсовик по алгоритмам и структурам данных}

## Булева логика

## Нечеткая логика

{НИР за третий семестр}

## Системы, основанные на правилах (экспертные системы)

## Конечные автоматы

### Недетерминированные конечные автоматы

[Шампандар]

### Вероятностные конечные автоматы

[Шампандар]

### Нечеткие конечные автоматы

[Шампандар]

## Списки действий

## Деревья поведения

[Buckland]

## Планировщики

[Buckland]

## Теория полезности

[Dave Mark]

## Проверка статистических гипотез

{НИР за третий семестр}

## Нейронные сети

[Шампандар]

## Генетические алгоритмы

[Шампандар]

# Цель и задачи работы

Целью данной работы является исследование методов моделирования принятия решений на примере ботов в компьютерной игре.

Задачи, решаемы в ходе работы:

* сформулировать задачи, которые должен решать бот;
* для каждой задачи выбрать методы и алгоритмы, обосновать выбор;
* реализовать выбранные алгоритмы;
* протестировать полученные реализации;
* сделать выводы об эффективности различных методов и алгоритмов для моделирования принятия решений.

# Выбор методов и алгоритмов для разработки

## Задачи, решаемые ботами, и подходящие для их реализации методы и алгоритмы

### Поиск пути

{Курсовик по алгоритмам и структурам данных}

### Прицеливание и стрельба

[Шампандар]

### Тактические решения

[Terrain Reasoning, Шампандар]

### Многоагентное взаимодействие

# Алгоритмизация и реализация

## Выбор и обоснование языка программирования и библиотек

Рассматриваемые методы и алгоритмы принятия решений не зависят от используемых технологий и языка программирования. В данной работе используется игровой движок Unity. Это один из самых популярных игровых движков {добавить ссылку на статистику}, позволяющий быстро создавать прототипы и полноценные игры. В качестве языка игровой логики в нем используются C# и javascript. В данной работе будет использоваться C#, так как он обладает более удобной объектной моделью и поддерживает сильную типизацию, которая позволяет уменьшить количество ошибок в ходе разработки.

## Интерфейс управления персонажем

Для управления персонажем используется следующий интерфейс:



Компонент, реализующий этот интерфейс, может принимать команды как от компонента, отвечающего за управление с помощью устройств ввода, так и от компонента, отвечающего за искусственный интеллект.

# Список использованных источников

1 Рассел С., Норвиг П. Искусственный интеллект: современный подход, 2-е изд. : Пер. с англ. – М.: Издательский дом «Вильямс», 2006.

2 Шампандар Алекс Дж. Искусственный интеллект в компьютерных играх. Как обучить виртуальные персонажи реагировать на внешние воздействия. : Пер. с англ. – М.: Издательский дом «Вильямс», 2007.

3 Buckland M. Programming game AI by example. – Wordware Publiching Inc. 2005.

4 Mark D. Behavioral Mathematics for Game AI. – Course Technology, a part of Cengage Learning, 2009.

5 Palacios J. Unity 5.x Game AI Programming Cookbook. – Packt Publishing, 2016.

6 Reynolds, C. W. Steering Behaviors For Autonomous Characters, in the proceedings of Game Developers Conference 1999 held in San Jose, California. Miller Freeman Game Group, San Francisco, California. Pages 763-782.

7 William van der Sterren. Terrain Reasoning for 3D Action Games. GDC 2001.