

ПЛАТФОРМА ДЛЯ ПРОГРАММИРОВАНИЯ ИГРОВОЙ СИСТЕМЫ

Галина Михайловна Рудакова, канд. физ.-мат. наук, профессор,

E-mail: gmrfait@gmail.com,

Дмитрий Олегович Ковригин, магистр,

Сибирский государственный технологический университет,

кафедра информационных технологий,

http://www.kit-sibstu.ru/

В статье идет речь о проектировании универсальной платформы для обучения и тестирования игровых интеллектуальных систем. Методы исследования, применяемые в работе, основаны на сравнении возможностей, которые предоставляют участникам существующие платформы для спортивного программирования. Предполагается, что спроектированная платформа, расширяющая список используемых языков программирования, может использоваться, в том числе, в целях популяризации соревнований по спортивному программированию.

Ключевые слова: проектирование, игровые интеллектуальные системы, спортивное программирование



Г.М. Рудакова

На сегодняшний день спортивное программирование является неотъемлемой частью ИТ-сообщества. Во всём мире ежегодно проводится множество событий так или иначе связанных с умением решать задачи в условиях ограниченного времени или ресурсов.

В области соревнований по спортивному программированию игровых интеллектуальных систем можно выделить Google AI Challenge [1] и российский аналог Russian AI Cup [2].

Google AI Challenge – является международным соревнованием, проводимым с 2009 года. Основная идея соревнования заключается в том, что на сервере определены правила игры, сервер выступает в качестве площадки для соревнований. Каждый игрок должен написать интеллектуальную систему, которая в соответствии с правилами будет осуществлять ходы до тех пор, пока в игре не наступит один из граничных результатов. Поддерживаемые языки, на которых можно присылать решения: C++, Java, Python, C#, JS, Ruby.



Д.О. Ковригин



Рисунок 1 – Распределение по использованию языков на соревнованиях по программированию игр Russian Ai Cup 2014

В 2010 году участие в соревновании приняли более 200 человек. В рамках данного соревнования было проведено 4 мероприятия, каждое из которых отличалось игрой, положенной в основу соревнования. Список представленных игр следующий: Rock Paper Scissors, Tron, Planet Wars, Ants.

Russian AI Cup – чемпионат по программированию искусственного интеллекта, организатором кото-

рого является Mail.Ru Group. Он проходит ежегодно, начиная с 2012 года. Принять участие могут все желающие, любых возрастов и квалификаций. Главное – иметь базовые навыки программирования [3]. Список возможных языков программирования ограничен, вы можете выбрать один из следующих вариантов: C++, Java, C#, Python, или Pascal. Чемпионат состоит из четырёх этапов: песочницы, двух отборочных раундов и финала. Фактически, участникам была предложена пошаговая стратегическая игра с формально определенными правилами и API для управления.

Если для некоторых соревнований это обусловлено жесткими правилами соревнований, так как это сделано в ACM-ICPC, то в соревнованиях связанных с созданием интеллектуальных систем, где такие правила не предусмотрены, ограничения обуславливаются техническими причинами, и связаны трудностью реализации для большого числа языков программирования. Особенно важной является информация, полученная по платформам, которые осуществляют тестирование и обучение игровых интеллектуальных систем.

Как показал анализ существующих платформ, детальная информация об их устройстве является закрытой, что обусловлено коммерческим характером мероприятий. Организаторы зачастую не заинтересованы в раскрытии технической информации, поскольку это может негативно сказаться на статусе мероприятия, за счёт заимствования конкурентами новых технологий. Таким образом, при анализе существующих платформ необходимо полагаться на данные о доступных языках программирования, предоставляемых организаторами.

Таблица 1

Языки программирования в игровых интеллектуальных системах

	Игровые Интеллектуальные системы		Другие	
	Russian AI Cup	Google AI Challenge	ACM-ICPC	Russian Code Cup
Доступные языки	C++, Java, C#, Python, Pascal	C++, Java, Python, C#, JS, Ruby	C, C++, Java	Java, Python, C/C++, C#, Perl, PHP

В ходе исследования было выявлено, что основной проблемой, затрудняющей поддержку большего числа языков программирования, является процедура обмена данными в ходе тестирования.

В сегодняшнюю игровую индустрию, ставшую чрезвычайно доходным бизнесом с многомиллионными бюджетами на игры, пошаговые стратегии вписываются плохо и находятся в состоянии стагнации. Новые игры выпускаются достаточно редко – пару игр в год. Легендарные же серии живут только благодаря своим фанатам. Время разработчиков-одиночек и маленьких студий прошло, игры делаются компаниями со штатом в сотни программистов и соответствующими расходами, пускаемыми в основном на огромное количество несущественных, для данного жанра, графических деталей.

Игровой искусственный интеллект – набор программных методик, которые используются в компьютерных играх для создания иллюзии интеллекта в поведении персонажей, управляемых компьютером. Игровой ИИ, помимо методов традиционного искусственного интеллекта, включает также алгоритмы теории управления, робототехники, компьютерной графики и информатики в целом. Возможности платформы, предоставляемые участнику игрового ИИ, заключаются в значительном размере блока управления, связанном с необходимостью обработки большого количества параметров и условий.

Существует мнение, что выражение «искусственный интеллект» в термине «игро-

вой искусственный интеллект» является преувеличением, поскольку игровой ИИ описывает не интеллект и использует немногие из направлений академической науки «Искусственный интеллект». В отличие от «реального» ИИ, конечной целью которого является вывод собственных правил, игровой ИИ [4, 5], как правило, состоит из наборов правил и эвристики, обычно этого вполне достаточно, чтобы предоставить игроку хороший геймплей, ощущения и впечатления от игры.

Важной особенностью игрового искусственного интеллекта является наличие большого блока управления. Это связано с тем, что практически в каждой системе, где используется игровой ИИ, перед ним стоит задача управления большим количеством параметров, которые зависят также от значительного количества условий.

Реализация ИИ сильно влияет на геймплей, системные требования и бюджет игры, и разработчики балансируют между этими требованиями, стараясь сделать интересный и нетребовательный к ресурсам ИИ малой ценой. Поэтому подход к игровому ИИ серьёзно отличается от подхода к традиционному ИИ – широко применяются разного рода упрощения, обманы и эмуляции. Например, с одной стороны, в шутерах от первого лица безошибочное движение и мгновенное прицеливание, присущее ботам, не оставляет ни единого шанса человеку, так что эти способности искусственно снижаются. С другой стороны – боты должны делать засады, действовать командой и т.д., для этого применяются «костыли» в виде контрольных точек, расставленных на уровне.

Игровой искусственный интеллект продолжает развиваться с целью достичь такого уровня, чтобы игрок был неспособен отличить компьютерного соперника от человеческого.

Увеличение понимания академического ИИ разработчиками игр и растущий интерес академического сообщества к компьютерным играм вызывает вопрос, насколько и в какой степени игровой ИИ отличается от классического. Однако существенные различия между различными прикладными областями искусственного интеллекта означают, что игровой ИИ всё ещё может быть рассмотрен как отдельная подотрасль ИИ. В частности, способность «законным» образом решить некоторые проблемы ИИ в играх через обман создаёт важное различие. Например, выведение позиции невидимого объ-



екта из прошлых наблюдений может быть трудной проблемой, когда ИИ применён к робототехнике, но в компьютерных играх неигровой персонаж может просто искать позицию в игровом графе. Такой обман может

Рисунок 2– Возможности платформы, предоставляемые участнику

привести к нереалистичному поведению, и поэтому не всегда желателен.

Игрок (участник) должен иметь возможность отправить написанную им интеллектуальную систему в платформу. После этого он может запустить бой с другой интеллектуальной системой, чтобы иметь возможность сравнить его систему с другими. Кроме того, необходимо учитывать потребность в визуализации процесса сражения, для этого необходимо предоставить участнику возможность просмотреть сражение.

Персонажей компьютерных игр, управляемых игровым искусственным интеллектом, делят на:

- неигровые персонажи – как правило, эти персонажи являются дружелюбными

или нейтральными к человеческому игроку;

- боты – враждебные к игроку персонажи, приближающиеся по возможностям к игровому персонажу; против игрока в любой конкретный момент сражаются небольшое количество ботов. Боты наиболее сложны в программировании;
- мобы – враждебные к игроку «низкоинтеллектуальные» персонажи. Мобы уничтожаются игроками в больших количествах ради очков опыта, артефактов или прохождения территории.

На основе сформированных требований выбрана технология обмена данными между интеллектуальными системами.

Поскольку весь обмен данными производится внутри платформы, то задержки между передачами сообщений не являются критичными, важно лишь обеспечить их синхронность для всех участников. Таким образом, исходя из полученных данных, выбор сделан в пользу протокола TCP [6], и весь обмен данными между интеллектуальными системами производится с его помощью.

Так как протокол TCP реализуется на уровне ядра системы, то его поддерживают большинство существующих на сегодняшний день языков программирования. Это обуславливает выбор данной технологии, положенной в основу реализации механизма обмена данными между игровыми интеллектуальными системами.

Механизм TCP предоставляет поток данных с предварительной установкой соединения, осуществляет повторный запрос данных в случае потери данных и устраняет дублирование при получении двух копий одного пакета, гарантируя тем самым, в отличие от UDP, целостность передаваемых данных и уведомление отправителя о результатах передачи.

Реализация TCP, как правило, встроена в ядро ОС, хотя есть и реализации TCP в контексте приложения.

Когда осуществляется передача от компьютера к компьютеру через Интернет, TCP работает на верхнем уровне между двумя конечными системами, например, браузером и веб-сервером. TCP осуществляет надежную передачу потока байтов от одной программы на некотором компьютере к другой программе на другом компьютере (например, программы для электронной почты, для обмена файлами). TCP контролирует длину сообщения, скорость обмена сообщениями, сетевой трафик.

Как указывалось, ранее, главной целью протокола TCP является обеспечение надежного, безопасного сервиса для логических цепей или соединений между парами процессов. Чтобы обеспечить такой сервис, основываясь на менее надежных коммуникациях, система должна иметь возможности для работы в следующих областях:

- базовая передача данных;
- достоверность;
- управление потоком;
- разделение каналов;
- работа с соединениями;
- приоритет и безопасность.

Возможность работы с TCP протоколом реализована во многих языках программирования, что делает его подходящим для использования в универсальной платформе.

Платформа должна обеспечивать участникам следующие возможности:

- участник имеет возможность отправить реализацию игровой стратегии;
- участник или система создают бой, который должен быть протестирован;
- бой тестируется, для чего запускаются процессы участников и движок игры;
- ход боя может быть просмотрен участником.

Игрок (участник) должен иметь возможность отправить написанную им интеллектуальную систему в платформу. После этого он может запустить бой с другой интеллектуальной системой, чтобы иметь возможность сравнить его систему с другими.

Кроме того, необходимо учитывать потребность в визуализации процесса сражения, для этого необходимо предоставить участнику возможность просмотреть сражение.

Компоненты предлагаемой платформы:

- Database – хранит состояние системы;
- TaskManager (таскменеджер) – берет из базы необработанные стратегии и игры, отдает задания воркерам и после ответа обновляет состояние стратегии/игры в базе;
- Worker (воркер) – получает задание на компиляцию, верификацию стратегии или тестирование игры, обрабатывает задание, возвращает ответ таскменеджеру. Для симуляции сражения воркер запускает game server и игровые стратегии;
- Game Server – запускается воркером для симуляции боя. Именно в этом приложении содержится логика игры;
- Worker's Cache – общий кэш для воркеров, хранит скомпилированные бинарники.
- Game Storage – хранилище для логов игр (необходимо для просмотра игр в браузере).

Для апробации платформы была выбрана игра, использованная на мероприятии Google AI Challenge 2010 под названием Трон (Tron). Данные, передаваемые в ходе апробации, имеют довольно простую структуру, что позволило упростить этап анализа и разработки платформы.

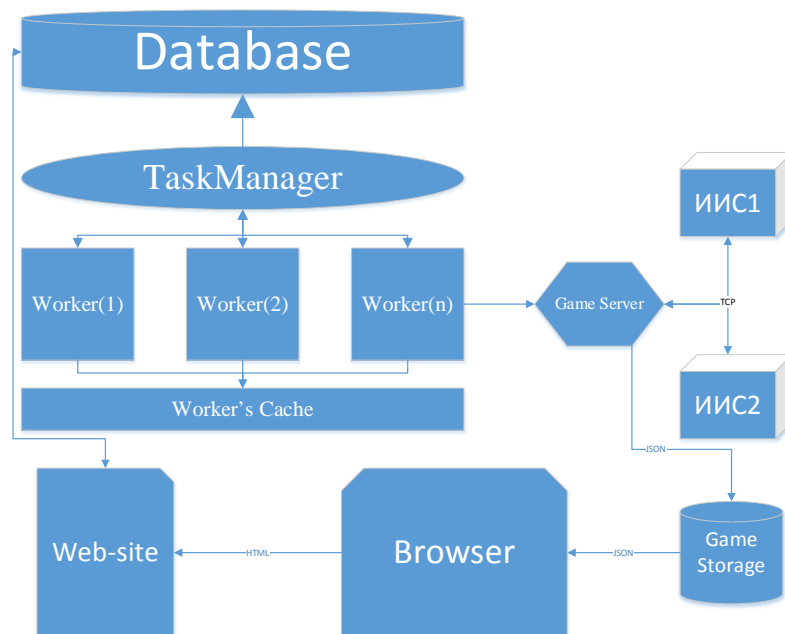


Рисунок 3 – Схема платформы

Литература

1. Правила Google AI Challenge Осень 2011 Graphics. URL: <http://habrahabr.ru/post/130979/>
2. Russian AI Cup 2014. URL: <http://devtype.blogspot.ru/2014/09/Russian-AI-Cup-2014-CodeHockey.html>
3. Кожевников Д.О., Рудакова Г.М. Причины и движущие силы поэтапной смены подходов внутриобъектно-ориентированной парадигмы программирования // Образовательные ресурсы и технологии. 2014*1. № 4. С. 68–78.
4. Создание своего домашнего игрового искусственного интеллекта. URL: <http://habrahabr.ru/post/193152/>
5. Основы игрового AI. URL: http://masandilov.ru/ai/basics_of_game_ai
6. Модели реализации протокола TCP и его перспективы. URL: <http://book.itep.ru/4/44/tcp.htm>

Programming platform gaming system

Galina Mikhailovna Rudakova, PhD, professor, Siberian State Technological University, Chair of Information Technologies.

Dmitry Olegovich Kovrigin, master, Siberian State Technological University, Chair of Information Technologies

In the article it is about universal design for learning and testing platform game of intelligent systems. Research methods used in the work, based on comparisons of features that provide participants with existing platforms for sports programming. It is expected that designed platform, expanding the list of programming languages can be used, including for the purpose of popularization of competitions on sports programming.

Keywords: Design, intelligent systems, gaming sports programming

УДК 004.053, 004.054

ГЕТЕРОГЕННАЯ КОМПЬЮТЕРНАЯ ИГРА КАК ОБРАЗОВАТЕЛЬНАЯ ТЕХНОЛОГИЯ

*Игорь Владимирович Соловьёв, д-р техн. наук, профессор, проректор по НИР,
E-mail: i.v.soloviev54@mail.ru,*

*Московский государственный университет информационных технологий, радио-
техники и электроники (МИРЭА/МГУПИ),
<https://www.mirea.ru>*

Раскрывается новый образовательный ресурс – гетерогенные деловые игры. Дается сравнение компьютерной деловой игры и гетерогенной деловой игры. Показано что гетерогенная компьютерная деловая игра решает большее количество задач и повышает качество образования. Описаны методические и технологические требования к данной технологии обучения. Показано, что данная технология обучения является поэтапной и на каждом этапе накапливает знания и информационные ресурсы. Раскрывается содержание гетерогенной компьютерной деловой игры.

Ключевые слова: игра, компьютерная деловая игра, гетерогенная компьютерная деловая игра, игровая модель, сценарий игры, игровое время, комплекс программно-технических средств обучения, жизненный цикл

Введение. Длительное время при автоматизации различных отраслей народного хозяйства, включая сферу образования, конкурировали две концепции. Первая заключалась в модернизации существующих технологий под средства и технологии автоматизации [1]. Вторая в преобразовании и придании этим технологиям нового качества за счёт использования средств автоматизации. Метод компьютерных деловых игр реализующий вторую концепцию и стимулирующий у обучаемых развитие системного и ситуационного мышления, является одним из современных инновационных инструментов совершенствования системы высшего образования России [2]. Анализ показывает, что этот метод постоянно развивается. Динамика развития привела к тому, что устоявшегося описания основ метода ещё не существует. Гетерогенное моделирование широко применяется при решении сложных задач [3]. В области компьютерных деловых игр этот подход пока не применяют. Поэтому актуальным представляется исследование гетерогенных деловых игр в синтезе с унифицированной программно-аппаратной платформой их реализации.