Студент: Иванов Иван Иванович, ВМИ-399

Руководитель: Петров Петр Петрович

Тема Разработка прототипа трехмерной компьютерной игры “Future Warfare” на базе платформы UNITY.

**Анализ предметной области и существующих работ по тематике курсового проекта**

1. **Предметная область проекта**

В данной работе рассматривается разработка компьютерной игры, позволяющей игроку вести боевые действия с модульными роботами. Способность изменять свою конфигурацию во время боя, предоставляет модульным роботам возможность быстро менять тактику ведения боя в зависимости от ситуации.

ТЕОРИЯ ГРАФОВ В АРХИТЕКТУРЕ МОДУЛЬНЫХ РОБОТОВ

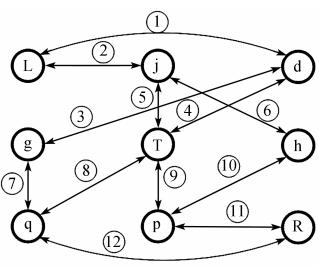
*Использование матрицы смежности*

Реконфигурируемые модульные роботы имеют множество конфигураций, адаптируясь к различным средам и задачам. Поиск нужной конфигурации усложняется тем, что возможное число конфигураций растет экспоненциально с увеличением числа модулей. В работе [4] для решения этой проблемы предлагают использовать матрицу смежности.

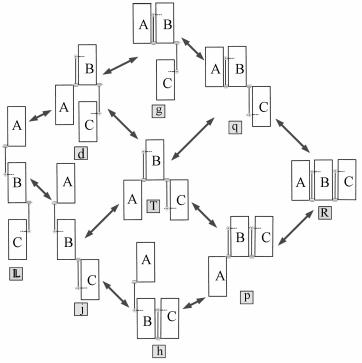
Смежные конфигурации можно перестроить друг в друга, в то время как не смежные конфигурации перестраиваются различными путями. Таким образом, конфигурация может быть представлена вершиной графа, а реконфигурация ребром графа. Такой граф может быть представлен матрицей смежности. Для примера возьмем модульного робота AMOEBA-I [4] (рисунок 1) и представим его конфигурации в виде графа (рисунок 2) (Более подробно эти конфигурации представлены на рисунке 3).



Рис. 1. Реконфигурируемый модульный робот AMOEBA-I [4]

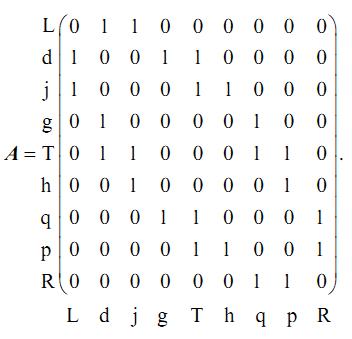


**Рис.** **2.** Представление конфигураций Модульного робота AMOEBA-I [4] в виде графа



**Рис.** **3.** Подробное представление конфигураций Модульного робота AMOEBA-I [4]

Для этого модульного робота существует 9 конфигураций и 12 типов смежных отношений между ними. Каждое смежное отношение предполагает, что одна конфигурация может быть преобразована в другую. Этот граф можно представить в виде матрицы смежности.



Данный способ позволяет хранить все конфигурации модульного робота в удобной форме, а алгоритмы из теории графов обеспечивают быстрый поиск и переход в нужную конфигурацию. Что позволяет модульному роботу, находящемуся в опасной для него среде, быстро перейти в новую конфигурацию, если старая конфигурация не может справиться с заданием.

*Использование матрицы инцидентности*

Модульных роботов сложно моделировать из-за динамического изменения их структуры. При изменении количества модулей в структуре изменяется количество связей и число степеней свободы, что влияет на кинематику робота. Поэтому необходимо использовать метод, легко представляющий конфигурацию структуры.

В работе [1] предлагают представить модульную систему соответствующим графом, показывающим отношение, существующее между различными модулями. Этого можно достичь путем маркировки вершин и ребер графа таким образом, чтобы по метке можно было, определить к какому типу относится, та или иная вершина или ребро. Для примера возьмем модульного робота RobMAT [1] и представим одну из его конфигураций (рисунок 4) в виде взвешенного графа (рисунок 5).

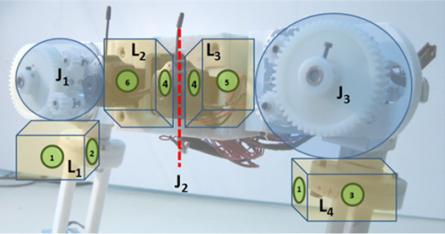


Рис. 4. Одна из конфигураций модульного робота RobMAT

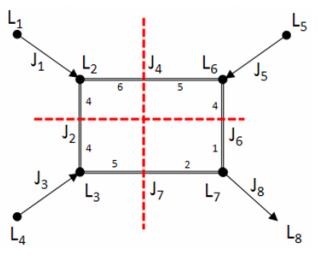
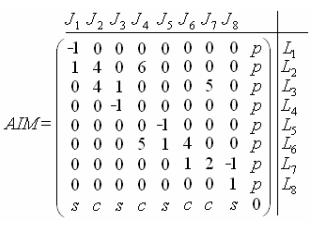


Рис. 5. Граф, представляющий конфигурацию модульного робота RobMAT [1]

В этом графе (рисунок 5) вершины представляют собой модули, а ребра связи между ними. Причем в эту конфигурацию входят связи и модули различных типов. Для представления такого графа в работе [1] используют матрицу инцидентности AIM, в которой последняя строка используется для определения типа связи, а последний столбец для определения тип модуля.



Таким образом, матрица AIM содержит достаточно информации, чтобы описать конфигурацию модульного робота с различными модулями и различными типами связей между ними. К тому же эту матрицу легко модифицировать при добавлении или удалении модуля. Это позволяет решить проблему описания конфигураций модульного робота с большим количеством различных типов модулей.

ИСПОЛЬЗОВАНИЕ МУЛЬТИАГЕНТНОЙ СИСТЕМЫ ДЛЯ ОРГАНИЗАЦИИ МОДУЛЬНЫХ РОБОТОВ

Модульные роботы могут содержать огромное количество различных модулей, каждый из которых мыслит самостоятельно. Для выполнения большинства задач им необходимо взаимодействовать друг с другом. Для решения этой проблемы в работе [3] предлагают использовать мультиагентную систему, т.е. можно представить модульного робота как мультиагентную систему, а каждый его модуль представить в виде агента этой системы.

Мультиагентная система – это система, содержащая группу агентов, которые могут взаимодействовать между собой.

Агент – нечто, что способно воспринимать свое окружение и изменять его своими действиями [8, 9]. Агент обладает следующими свойствами:

1. реактивность;
2. автономность;
3. целенаправленность;
4. непрерывность функционирования;
5. коммуникативность;

Свойство реактивности означает, что агент временами отвечает на изменения в окружении. Агент может перевести окружение из некоторого состояния в некоторое другое состояние, но не из любого в любое.

Свойство автономности означает, что агент сам контролирует свои действия.

Свойство целенаправленности означает, что у агента имеется определенная цель и его поведение подчинено этой цели.

Свойство непрерывности функционирования означает, что агент постоянно находится в состоянии исполнения.

Свойство коммуникативности означает, что агент может общаться с другими агентами, используя для этого некоторый язык.

Информация, которой обладает система агентов, является обычно распределенной. Агенты могут следить за окружением из разных положений и получать информацию в различные моменты времени. Таким образом, состояние окружения является частично обозримым для каждого агента.

Каждый агент для выполнения своего задания может выбрать себе роль исходя из своих возможностей. Играя каждый свою роль, агенты общаются между собой: координируют свои действия, обмениваются знаниями, ведут переговоры.

Таким образом, представив модульного робота как мультиагентную систему можно организовать успешное взаимодействие его модулей между собой.

1. **Анализ аналогичных проектов и существующих решений для реализации проекта**

ИНСТРУМЕНТЫ ДЛЯ РАЗРАБОТКИ ИГРЫ

Для создания качественной трехмерной компьютерной игры необходимо использовать качественные инструменты. Одним из таких инструментов является Unity [5]. Он содержит огромное количество технических средств для создания компьютерных игр.

1. Встроенный физический движок Ageia PhysX™
2. Высоко оптимизированный графический "пайплайн" как для DirectX, так и для OpenGL
3. Потоковое аудио и видео
4. Поддержка трех языков скриптования: JavaScript, C#, Boo

Игровой движок Unity3D полностью интегрирован в Среду Разработки Unity. Тесная интеграция позволяет прямо в редакторе получить всё то, на что будет способна выпущенная игра.

1. **Заключение**

В результате анализа источников, нами была выбрана платформа Unity для реализации проекта игры позволяющей игроку вести боевые действия с модульными роботами.

**Литература**

1. [Baca](http://www.springerlink.com/content/?Author=Jos%c3%a9+Baca) [J.](http://www.springerlink.com/content/?Author=Jos%c3%a9+Baca), [Yerpes](http://www.springerlink.com/content/?Author=Ariadna+Yerpes) [A](http://www.springerlink.com/content/?Author=Ariadna+Yerpes)., [Ferre](http://www.springerlink.com/content/?Author=Manuel+Ferre) [M](http://www.springerlink.com/content/?Author=Manuel+Ferre)., [Juan A.,](http://www.springerlink.com/content/?Author=Juan+A.+Escalera) [Aracil](http://www.springerlink.com/content/?Author=Rafael+Aracil) [E](http://www.springerlink.com/content/?Author=Juan+A.+Escalera)., [Aracil](http://www.springerlink.com/content/?Author=Rafael+Aracil) [R](http://www.springerlink.com/content/?Author=Rafael+Aracil). Modelling of Modular Robot Configuration Using Graph Theory, [Lecture Notes in Computer Science](http://www.springerlink.com/content/0302-9743/) // Hybrid Artificial Intelligence Systems. Vol.5271. 2008. pp. 649-656.
2. [Busquets](http://www.springerlink.com/content/?Author=D%c3%addac+Busquets) D., [Sierra](http://www.springerlink.com/content/?Author=Carles+Sierra) [C](http://www.springerlink.com/content/?Author=Carles+Sierra)., [López R.](http://www.springerlink.com/content/?Author=Ramon+L%c3%b3pez+de+M%c3%a0ntaras) A Multiagent Approach to Qualitative Landmark-Based Navigation // Autonomous Robots. Vol.15. №2. 2003. pp. 129-154.
3. Carrillo E., Duhaut D. Methods for Collective Displacement of Modular Self-reconfigurable Robots // Climbing and Walking Robots. Vol.10. 2006. pp. 641-648.

1. [JinGuo Liu](http://www.springerlink.com/content/?Author=JinGuo+Liu), [YueChao Wang](http://www.springerlink.com/content/?Author=YueChao+Wang), [Bin Li](http://www.springerlink.com/content/?Author=Bin+Li), [ShuGen Ma](http://www.springerlink.com/content/?Author=ShuGen+Ma) and [DaLong Tan](http://www.springerlink.com/content/?Author=DaLong+Tan). Center-configuration selection technique for the reconfigurable modular robot // Science in China Series F: Information Sciences, Vol.50. №5. 2007. pp. 697-710.
2. Will Goldstone. Unity Game Development Essentials. 2009. 316 p.
3. Арлоу Д., Нейштадт И. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование. 2-е издание. СПб: Символ-Плюс, 2007. 624 с.
4. Интеллектуальные агенты. URL: <http://www.itfru.ru/index.php/intellectual-agents>. (дата обращения 14.3.2012).
5. [Миков А.И.](http://www.intuit.ru/lector/284.html), [Замятина Е.Б.](http://www.intuit.ru/lector/285.html) Распределенные интеллектуальные системы на основе агентов // [Распределенные системы и алгоритмы](http://www.intuit.ru/goto/course/distrsa/). 2008. URL: <http://www.intuit.ru/department/algorithms/distrsa/10/> (дата обращения 22.12.2011).
6. Стюарт Р., Питер Н. Искусственный интеллект: Современный подход // Интеллектуальные агенты. 2-е издание. М.: Вильямс, 2006. С. 75-109.
7. Шампандар А. Искусственный интеллект в компьютерных играх. М.: Вильямс. 2007, 768 с