

УДК 004.896

СИСТЕМЫ МОДЕЛИРОВАНИЯ АЛГОРИТМОВ ПОВЕДЕНИЯ И ДВИЖЕНИЯ АВТОНОМНЫХ МОБИЛЬНЫХ РОБОТОВ

К.т.н. А.А. Левтеров¹, Ю.А. Нечитайло²

1. Национальный университет гражданской защиты Украины, г.Харьков

2. Харьковский национальный автомобильно-дорожный университет

В работе описаны системы моделирования поведения роботов с открытым кодом а также системы имитационного моделирования работы роботов в условиях различных сред. Проведен анализ программных средств и их оптимального использования.

У роботі описані системи моделювання поведінки роботів з відкритим кодом, а також системи імітаційного моделювання поведінки роботів в умовах різних середовищ. Проведено аналіз програмних засобів та їх оптимального використання.

In the work systems of modeling of behavior of robots with an open code and also systems of imitatsonny modeling of operation of robots in the conditions of various environments are described. The analysis of software and their optimum use is carried out.

Ключевые слова: робототехника, мобильный автономный робот, алгоритмы поведения, алгоритмы движения

Успешное внедрение промышленных роботов в процесс автоматизированного производства различной продукции в настоящее время привело к возможности, а также необходимости, научных исследований в области создания особых мобильных роботов. Нужны роботы, которые могут без помощи оператора тушить пожары, самостоятельно передвигаться по заранее неизвестной реальной пересеченной местности, выполнять спасательные операции во время стихийных бедствий, аварий атомных электростанций и т.п.

Для создания таких робототехнических систем, несмотря на накопленный опыт, большинство разработок мобильных роботов испытывают дефицит алгоритмов и моделей, имитирующих движение и поведение робота в условиях окружающей среды, а также программ для решения наиболее трудной проблемы – автоматического управления траекторией для достижения цели в присутствии большого числа заранее неизвестных помех движению робота. Здесь наиболее важными факторами является маневренность робота, связанная с конструктивными особенностями движителя, сенсорное обеспечение восприятия окружения и алгоритмы принятия двигательных решений.

Перечисленные обстоятельства обосновывают актуальность решаемой задачи разработки алгоритмов и их компьютерных имитационных моделей, позволяющих на стадии разработки оценить эффективность работы аппаратно-программных средств и систем формирования сигналов управления автоматическим движением мобильного робота.

Среды моделирования предоставляют виртуальные арены для тестирования, измерения и визуализации алгоритмов роботов без больших затрат ресурсов (и

времени) процесса разработки. Данная работа посвящена некоторым из open source-инструментариев (с открытым исходным текстом) роботов под Linux и Windows, продемонстрирует их возможности, а также поможет решить какой из них лучше подходит для данных целей.

Рассмотрим открытые инструментарии для Linux. Существует несколько доступных открытых инструментариев для построения роботизированных систем управления.

Открытый Динамический Движок (Open Dynamics Engine - ODE) [1], представляющий собой движок физики с открытым исходным кодом, с помощью которого можно моделировать динамику тел с шарнирными сочленениями. Таким образом, можно моделировать физику объектов реального мира вне зависимости от графических библиотек (в качестве которой можно использовать OpenGL). В добавление к этому, ODE поддерживает средства обнаружения наложений объектов для процесса моделирования в реальном времени.

В частности, среди перечисленных возможностей ODE поддерживает такие типы сочленений, как шарнирные, створчатые, угловые, фиксированные, а также полозья и двойные шарниры (для соединений в средствах передвижения). Он также поддерживает различные простейшие наложения (как, например, сферы и плоскости) и несколько промежутков между наложениями. Если моделировать мобильный робот или беспилотное летательное средство передвижения в реалистичных средах, то ODE, вместе с программным интерфейсом приложения (API) OpenGL, позволит генерировать как фотореалистичную графику, так и реалистичную физику.

Программа Simbad [2] представляет собой программу моделирования трехмерных роботов и изучения алгоритмов искусственного интеллекта (AI) в контексте создания автономных роботов, и включает в себя качественный графический пользовательский интерфейс (GUI) для визуализации действий робота, который также позволяет "видеть" окружающий мир с точки зрения робота. Несмотря на то, что разработка в Simbad является простой, моделирование робота предполагает достаточно большой объем работы.

Программа моделирования Simbad является отличной средой для тестирования идей в алгоритмах интеллектуальных роботов.

Программа моделирования роботов TeamBots представляют собой портативную программу моделирования мультиагентных (multi-agent) роботов, которая позволяет создавать мультиагентные системы управления в динамических средах с визуализацией. В сравнении с программой моделирования Simbad

Технология приборостроения

программа TeamBots является уникальной, благодаря возможности создавать переносимую систему управления и реализовать ее в программе моделирования, а затем протестировать в реальном мобильном роботе (используя робот Nomadic Technologies Nomad 150).

Интерфейс API программы TeamBots [3] предусматривает абстрактный слой для системы управления (см. рис. 1). В результате чего, системе управления совершенно не важно, где именно ее запускают: в программе моделирования в искусственной среде (TBSim) или на платформе мобильного робота в реальной окружающей среде (TBHard).

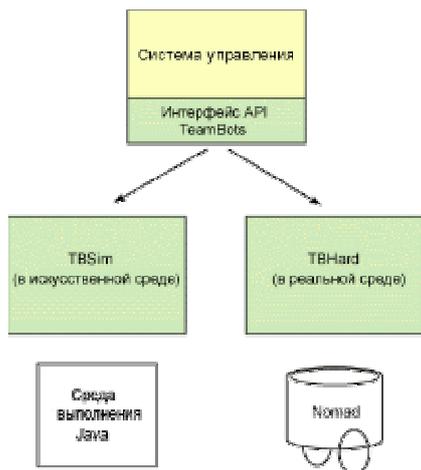


Рис. 1. Абстрактный слой для системы управления в интерфейсе API TeamBots

Среда моделирования TeamBots позволяет гибкую настройку и легкое построение искусственных сред с объектами и другими роботами. Вместе с тем, объекты не обязательно должны быть статичными. Можно поместить объекты, которые могут двигаться по всей среде, или объекты, которые могут двигаться в случае, когда их толкнул робот (например, мяч).

С помощью TeamBots можно создавать модели роботов различных типов. TeamBots представляет собой отличную среду как для разработки, так и моделирования мобильных роботов, а также для реализации их в реальных роботах, используя среду TBHard. TeamBots является программой с открытым, поэтому она может свободно использоваться в образовательных и исследовательских целях. Программа моделирования была разработана на языке Java и поставляется с полным исходным кодом и рядом примеров.

В приложении ROBOT RAD [2] можно осуществить отладку работы разработанных видео сенсоров мобильных роботов и целесообразность их применения. При помощи данной программы можно создавать модели электронной схемы модулей робота, имитирующих аппаратную часть, которая выглядит следующим образом (см. рис. 2)

Как видно из рис.2 в ROBOT RAD возможно создавать множество блоков, выполняющих различные функции при работе мобильного робота. Данная система идентична системе Gazebo под Linux, но в отличие от последней не распространяется с открытым кодом.

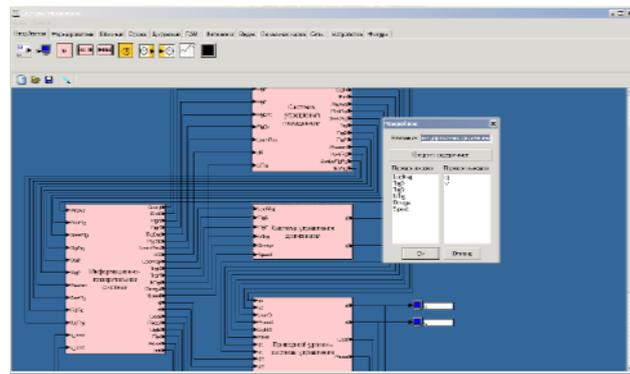


Рис. 2. Взаимосвязи модулей робота в ROBOT RAD

Еще одна система, разработанная в МВТУ им. Баумана [4], представляет набор блоков, математических моделей и алгоритмов, специализированных для моделирования динамических объектов, в том числе и мобильных роботов. Пример работы имитационной модели стационарного робота показан на рис.3.

В данной программе был реализован алгоритм следящей системы для управления тележкой робота, который приведен на рис.3.

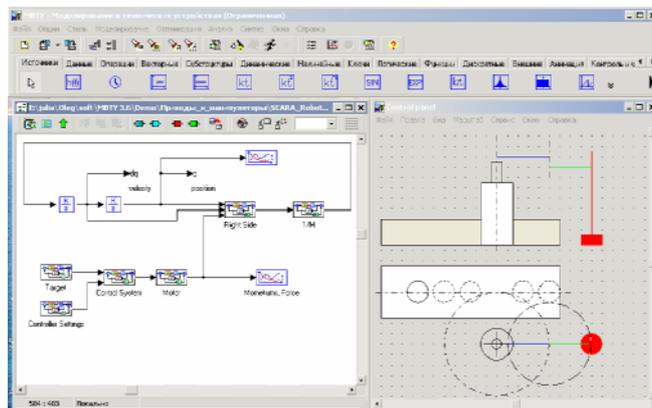


Рис. 3. Имитационная модель работы промышленного робота

Приложение MobaSim [5] для отработки моделей сенсорного восприятия робота, а также его правильного поведения при преодолении различной сложности препятствий можно использовать программу MobaSim.

В рамках этой статьи предложен пример имитационной модели поведения мобильного робота с двумя сенсорами с использованием адаптивной системы навигации. Пример работы программы приведен на рис.4. Красная линия показывает траекторию движения робота к желтой точке назначения. Траектория формируется в зависимости от положения робота по отношению к точке назначения и расположению препятствий в виде геометрических фигур. В программе MobaSim также можно использовать различные алгоритмы поведения как одного, так и целой группы роботов, имеющих разные параметры навигационных систем и сенсоров.

Процесс построения модели Simulink представляет собой компоновку модели и задание необходимых параметров.

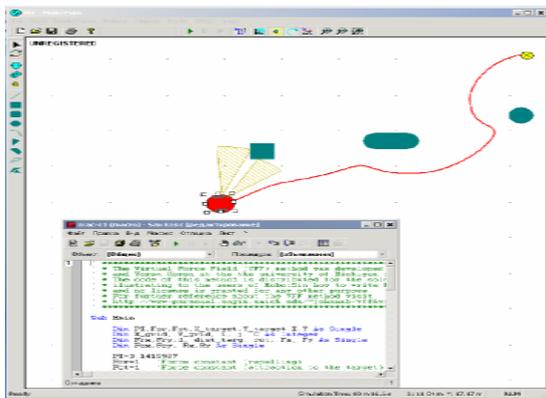


Рис. 4. Работа MobaSim с примером программного кода

Ввиду того, что в данной статье рассматривается только модель системы навигации и управления, подробное описание остальных систем робота не приводится.

Рассмотрим имитационную модель дискретно-позиционной системы управления робота [7] в системе MatLAB с использованием встроенного пакета Simulink. Выделив основные модули систем робота, получим схему взаимосвязанных модулей, представленную на рисунках 5 и 6. Входные сигналы (кроме сигнала управления) объединяются мультиплексором и подаются на вход функциональной подсистемы анализа рассогласования текущей и терминальной координат. Сигнал управления поступает на устройство сравнения, анализирующее включение или выключение процедуры опроса датчика.

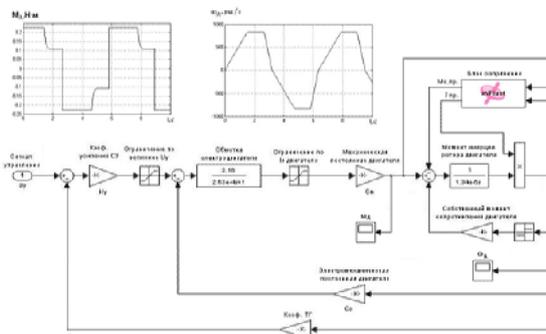


Рис. 5. Имитационная модель системы управления рабочим органом робота

Таким образом, полный цикл работы робота может быть смоделирован последовательным соединением блоков-подсистем.

Проведя анализ моделей, предложенных в [6-11], и опираясь на вышеизложенное, можно сделать вывод, что имитационное моделирование механических систем мобильных роботов удобно проводить в пакете MATLAB, содержащем в своем составе инструмент визуального моделирования SIMULINK, который обладает всеми необходимыми для этого инструментальными средствами и высокой скоростью вычислений.

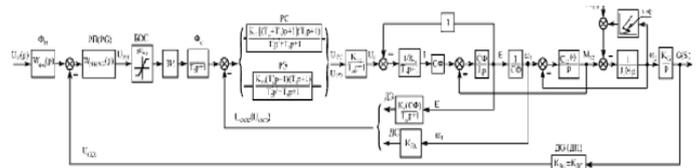


Рис. 6. Подробная схема системы управления тележкой робота

Для моделирования же систем навигации и зрения, поведения и др., а также электронной схемы роботов удобнее использовать приложения RAD или MobaSim, API TeamBots, ODE. Эти приложения имеют ряд ограничений. Приложения, такие как: Gazebo, Carmen и др. [2].

Наибольшее число программ, в том числе Gazebo и Carmen, для моделирования систем роботов используют операционную систему Unix, что приводит к ограничению их массового использования.

Таким образом, для решения проблем, связанных с моделированием робототехнических систем, необходимо разделять системы поведения, навигации и работу отдельных модулей (алгоритмы обработки данных). Т.е. для моделирования поведения роботов визуальные приложения, где будут видны ошибки, не учтенные при проектировании систем поведения и навигации. Для отладки электронной составляющей лучше использовать приложения типа Simulink.

СПИСОК ЛИТЕРАТУРЫ:

1. Russell Smith. *Open Dynamics Engine*. – 2014.–773с.
2. Инструментарий роботостроения с открытым исходным кодом. М. Тим Джонс, инженер-консультант, <http://www.ibm.com/developerworks/linux/library/l-robotools/index.html>. ((дата обращения: 05.09.2014 г.)
3. API программы TeamBots <http://www.teambots.org> (10.09.2014)
4. Программный комплекс. «Моделирование в технических устройствах» (ПК «МВТУ»)/ <http://mvtu.power.bmstu.ru/> (10.09.2014 г.)
5. MobaSim. http://www.mobotsoft.com/?page_id=9/ (12.10.2014 г.)
6. Добрынин Д.А., Карпов В.Э.. Моделирование некоторых форм адаптивного поведения интеллектуальных роботов// *Информационные технологии и вычислительные системы №2*, 2006 с.45-56
7. Кисельков А.И. К построению математических моделей роботов// *Машиноведение*. 1989. №4, с. 68 – 73.
8. Леоненков А.В., *Нечеткое моделирование в MATLAB и fuzzyTECH*.//Санкт-Петербург: - «БХВ-Петербург»,2005.
9. Маслов О., Пузанов А., Куванов К., Платов О. *Проектирование и изготовление высокопроходимых мобильных роботов специального назначения с использованием современных САПР*. – *CAD/CAM/CAE Observer*, 2005. №2(20), 3(21).
10. Цюй Дунъюэ. *Управление мобильным роботом на основе нечетких моделей*. "Современные проблемы науки и образования" №6 часть 2, 2007.
11. Yoshinori Kuno, Nobutaka Shimada, Yoshiaki Shirai. *A Robotic Wheelchair Based on the integration of Human and Environmental Observations: Look Where You're Going*. – *IEEE Robotics and Automation Magazine*, March, 2003. – Pp.26-34.