ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 5, September 2013

Mobile Robots: Successes and Challenges in Artificial Intelligence

Jitendra Joshi (Research Scholar), Keshav Dev Gupta (Assistant Professor), Nidhi Sharma (Assistant Professor), Kinnari Jangid (Assistant Professor)
1# Department of Computer Science and Engineering
Jayoti Vidyapeeth Women's University,
Vedaant Gyan Valley, Village Jharna, Mahala - Jobner, Link Road,
Jaipur Ajmer Express Way, NH-8, Jaipur, Rajasthan, (India)-303007
2, 3, 4# Maharani Girls Engineering College
N.H.-8, Near Mahindra World City (SEZ), Ajmer Road, Jaipur, (Rajasthan) – 302037

Abstract— These research papers provide mobile robot engineering encompasses techniques from a wide variety of scientific and engineering domains. When design a robot, the whole system should be considered. This allows the developer to take into account all the factors that influence the robot itself. In this research paper used of robot system design architecture components and also used of robotic environments.

***Index Terms*— Mobile Robot, Design, Framework, Robot Architecture.**

I. INTRODUCTION

Mobile robot engineering encompasses techniques from a wide variety of scientific and engineering domains. Electronics, mechanics, computer science, biology, chemistry, physics and psychology have all played a significant role in what is now referred to as robotics. With such a diverse background it is difficult to define the term 'robot'. The problem is confounded by ambiguous descriptions and the overlapping of 'Robotics' with similarly unbounded scientific domains, such as 'Cybernetics' and 'Artificial Intelligence'. Intelligent [1], Autonomous and Unmanned are descriptions that immediately confuse not only the engineers of robot systems, but also the wider public. Mobile robotics is becoming an increasingly popular field of research, especially as embedded computing technology matures and gets ever more sophisticated. The goal of a mobile robot or a swarm of mobile robots is typically to travel through an unknown environment autonomously, while being continuously aware of both its surroundings and its position in relation to those surroundings. Mobile robots, such as wheeled robots and unmanned underwater vehicles, use a form of robot architecture. The term 'robot architecture' is commonly used to describe the software structure and its action selection methods. The robot architecture provides the robot command structure and has a wide-ranging effect on the robots ability to perform its desired tasks. Further to traditional control systems, a robot's architecture may be capable of performing deliberative actions. Traditionally, robot architectures are constrained to cognition and interaction with the vehicle. However, this paper will use a unified design framework to describe the overall operation and structure of the robot system, which may include other robots, users and environments and various action selection techniques [2]. Whereas traditionally, robot architectures have focused on abstraction of hardware and software elements, the proposed unified approach does not attempt to impose such boundaries.

II. ROBOT SYSTEMS ENGINEERING

When designing a robot, the whole system should be considered. This allows the developer to take into account all the factors that influence the robot itself [3]. The developer should be concerned with the functionality that the system should exhibit, not necessarily if it is located in hardware or software. A robot design framework should therefore encompass a meta-model for the entire system, rather than simply encompassing the software aspects within the robot. When designing a robot system, the designer should consider many other aspects, including Robot Architecture, Control Systems, Self Invariance, Learning, Centralized and Distributed Processing, Multi-threading, Shared Resources, Robustness, Reliability, System Decomposition, Top-Down and Bottom-up design, Component re-use, Open and Closed Systems and Robot Software Development Tools.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 5, September 2013

III. Robot System Components

A number of common components exist within a robot system. The generic component categories in Table 1 have been identified by examining a large number of robot architectures and their functionality [4]. A detailed analysis of each component can be found in following table.

Table 1: Generic robot system components

Component	Description
Perception	Perception of the environment provided by sensors and other robots.
Planning	Planning to optimize the selection of, and route between goals.
Guidance	Guidance to determine the current location on a plan and provide the next action.
Management	Management to manage the activation and de-activation of behaviors and components.
Action Selection	Action Selection providing decision making about the most appropriate action.
Human Robot Interaction (HRI)	Human Robot Interaction (HRI) providing motivation to the robot and feedback to the robot operators.
Motivation	Motivation allowing a robot operator to provide the robot with a set of objectives.
Behavior	Behavior providing one or more actions when provided with stimulus from vehicle sensors or perception.
Actuator	Actuators creating vehicle or manipulator motion.
Communication	Communications to communicate between distributed components, robots and HRI devices.
Chassis	Chassis providing the hardware framework of the robot.
Processor	Processor to perform processing activities.
Power	Power to provide power to the robot components.
Payload	Payload to allow the robot to transport objects or tools, such as other robots or remotely deployed sensors.

A. Design System

The unified design framework is split into dynamic and static models. Dynamic models express the flow of control and data between common activities. Static model class diagrams are used to describe the overall framework and structure in which components reside. Furthermore, new techniques can be added to the class structure as and when they become available.

B. Dynamic Modeling

Section 1 has identified three architecture paradigms; namely deliberative, reactive and hybrid. This work has used the UML activity diagrams to illustrate flow of data and control amongst the main components of these three paradigms using common components, such as sense, plan and act. Table 2 describes the design patterns commonly found within robot systems. These design patterns can be used to build the architectures analyzed in robotic model. These design patterns may then be used to develop new architectures using the unified framework classes of the static model. It is our intention to expand upon these existing design patterns to include other generic robot system processes, such as SLAM and hierarchical planning.

Table 2: Robot architecture design patterns and it's detailed of design patterns are provided within following table

Design Pattern	Description
Sense-Act (SA)	Used to directly link sensor data to actuators.
Sense-Decide-Act (SDA)	Used to determine an appropriate action, based upon a sensor value.
Sense-Plan-Act (SPA)/ Sense-Model-Plan-Act (SMPA)	Used to describe the process of creating and executing a new plan based upon sensor information.
Parallel Sense-Plan-Act (SPA)	Allowing Sense, Plan and Act components to act in parallel, at potentially differing frequencies.
Action Selection (Trigger based)	Selecting an appropriate action from a set of pre-defined actions.
Action Selection (Action based)	Selecting an appropriate action from a set of newly created actions.
Repair-based planning	Repairing an existing plan to accommodate a change in the robot or environment.
Guidance	Determining the position and current action within a plan.
Three-level architecture (Trajectory style)	Creating and following an appropriate trajectory based upon sensor inputs.
Three-level architecture (Management style)	Creating and following a sequence of controller activations based

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 5, September 2013

	upon sensor inputs.
Hybrid Three-level architecture	The combination of three-level architecture with action selection and behaviors to form a reactive and deliberative system.

C. Static Modeling

Static Modeling is used to define the structure of the robot system. This structure forms the backbone of the unified framework. The static models use UML classes to represent each component of the system. Each component may be subdivided into further components or types. The following sections describe the base classes of the framework. It is intended that any modifications or additions to the methods or parameters of the base classes will result in a derived class. This Open-Close Principle (OCP) approach enables the framework to be expanded whilst still allowing for backwards compatibility.

D. Robotic System

A robot system can, depending upon the designer's point of view, be described as:

1. The systems and components contained within a robot, or
2. The system in which the robot exists.

This unified approach will accommodate both descriptions. A robot system can therefore consist of environments, robots, users and interaction tools. Users may interact with robots and environments using interaction tools. Examining the entire system enables the developer to treat the robot as a team member, rather than a subservient agent. Furthermore, examining the system in which the robot exists allows collective, multiple or social robot development to be integrated within the same framework.

IV. ROBOTIC ENVIRONMENT

The following sections describe the Environment, User, Interaction and Robot classes.

A. Environment

The Environment class exists within the robot system (i.e. the real world) and the robot's perception i.e. the robot's world map. The environment may be real or virtual i.e. may exist within software.

In the virtual environment, or robot perception, it is necessary to use a representation [5]. This representation is a replica of the real environment and is used for simulation and/or planning. Common representation types include grids, graphs, surfaces, databases and semantic maps. Some or all of these representations may be used to describe the environment.

B. User

The User may interact directly with a Robot by sensing its position or manipulating its configuration. The User may also, more commonly, interact with the Robot using an Interaction device. The User will provide, or be provided with the mission requirements; it is then their responsibility to convert the mission requirements into an appropriate form using an appropriate Interaction device [6].

C. Interaction

The user may instruct the robot using touch, speech, facial expressions or movements using a Human Robot Interaction (HRI) device. The robot may respond through movement or though an HRI device, such as a graphical displays [7]. The HRI device is used to define the robot's motivation or control its actuators. Motivation consists of a plan. Once a plan has been created it may be communicated to the robot using a form of communication contained within the Communication class [8]. The plan may comprise high-level goals, which the robot may process in order to form a planned path.

D. Robot Class

The robot class is described in table 3. A robot consists of zero or one vehicles, associated with zero or more cognitive components. Common robot variants include: A single vehicle without cognition [9].

For example: - A remote control car provides methods to communicate with and control the vehicles actuators. The cognition is embedded within the user.

A single vehicle with cognition (embodied intelligence).

For example: A robot with a level of on board autonomy. This on-board autonomy may relate to the ability to plan a route.

In which case, the developer should create multiple robot instances, most of which contain only vehicles and one of also contains a cognitive component [10].

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 5, September 2013

Table 3: Vehicle classes of robotics system

Class	Description
Actuator	Provides functionality to move the robot within or interact with the robot's environment. Internal to the robot, the actuator may access sensor information for low-level feedback.
Sensor	Provides functionality to access sensor information. This sensor information may be provided on a continuous stream or polled.
Beacon	Provides a signal to other robots, or may be used by active sensors, such as sonar.
Dynamics /Kinematics Model	Describes the behavior of the robot when provided with actuator demands. This dynamics/kinematics model can be used for path planning, simulation or fault detection.
Communication	Allows the robot to send and receive plans or perceptions. The communications class can also include higher-level middleware.
State	Describes the robot's position and orientation within its environment.
Processor	Provides the processing and memory components available on the robot.
Payload	Provides the robot the capacity to transport other robots (including remote sensors) and users.
Power	Provides energy sources and storage to run the vehicle components.

V. CONCLUSION

In this paper, we have find out that mobile robot engineering encompasses techniques from a wide variety of scientific and engineering domains. The goal of a mobile robot or a swarm of mobile robots is typically to travel through an unknown environment autonomously. When designing a robot system, the designer should consider many other aspects, including Robot Architecture and number of common components exist within a robot system.

REFERENCES

[1] David Kortenkamp, R. Peter Bonasso, and Robin Murphy, editors. Artificial Intelligence and Mobile Robots. AAAI Press, 1998.

[2] J. Pack, A. M. Mankowski, and G. J. Freeman. A fire fighting robot and its impact on educational outcomes. Computers in Education, 9(3):2-5, July-September 1999.

[3] Gat, E. (1997) on three-layer architectures. Artificial Intelligence and Mobile Robots. MIT/AAAI Press.

[4] Murphy, R. (2000) Introduction to AI Robotics. MIT Press.

[5] Holt, J. (2001) UML for systems engineering. ISBN 0 85296 105 7, IEE 2001.

[6] Goodwin, J., Winfield, A., Zhu, Q. (2003) Towards a Generic Architecture for Autonomous Landing Systems. Proc. Toward intelligent mobile robot (TIMR, 03), Bristol, August 2003.

[7] Aha, D.W., Marling, C., and Watson, I. (Eds.) (2005).*The Knowledge Engineering Review, special edition on case-based reasoning*, volume 20 (3). Cambridge University Press.

[8] Buchanan, B.G. (2005). A (very) brief history of artificial intelligence. *AI Magazine*, 26(4): 53-60.

[9] Anderson, M. and Leigh Anderson, S.L. (2007). Machine ethics: Creating an ethical intelligent agent. *AI Magazine*, 28(4):15-26.

[10] Goodwin, J. (2008) A Unified Design Framework for Mobile Robot Systems. PhD Thesis. Bristol Institute of Technology, University of the West of England, October 2008. Available from http://www.AutonomousSolutions.co.uk.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 5, September 2013

AUTHOR BIOGRAPHY

Jitendra Joshi, B.Sc. M.C.A. M. Tech. Ph. D (Pursuing), is an Assistant Professor of computer science and engineering, Jayoti Vidyapeeth Women's University, Jaipur, Rajasthan, India. He has over seven years of experience in teaching computer science and engineering to graduate and post graduate students. He is member of IAE, IAE-Societies, IAS, CSI, IEEE, Computer Society of Australia and Computer Society of Singapore. He has published two books viz Data Structure and Algorithm, Computer Architecture and Micro Processor. He has published many research papers in his field of interest, viz. Algorithms, Artificial Intelligence, Neural Network, E-Commerce, Sensor Network and Network and Security.

Keshav Dev Gupta, M.C.A. M. Tech, is an Assistant Professor of computer science and engineering, Maharani Girls Engineering College Jaipur, Rajasthan, India. He has over seven years of experience in teaching computer science and engineering to graduate and post graduate students. He is member of IAE, IAE-Societies, and IAS. He has published two books viz OOPs with C++ and OOPs with JAVA programming. He has published many research papers in his field of interest, viz. Sensor Network, Network and Security.

Nidhi Sharma, B. Tech, M. Tech. (Purs.) is an Assistant Professor of computer science and engineering, Maharani Girls Engineering College Jaipur, Rajasthan, India. He has over two years of experience in teaching computer science and engineering.

Kinnari Jangid, B. Tech, M. Tech. (Purs.) is an Assistant Professor of computer science and engineering, Maharani Girls Engineering College Jaipur, Rajasthan, India. He has over two years of experience in teaching computer science and engineering.