Review

A review of control architectures for autonomous navigation of mobile robots

D. Nakhaeinia¹, S. H. Tang¹, S. B. Mohd Noor¹, and O. Motlagh¹

¹Faculty of Engineering, University Putra Malaysia, 43300 UPM, Serdang, Selangor, Malaysia.
²Institute of Advanced Technology (ITMA), University Putra Malaysia, 43300 UPM, Serdang, Selangor, Malaysia.

Accepted 05 January, 2011

A mobile robot as an intelligent system needs to sense the surroundings, perceive the working environment, plan a trajectory and execute proper reaction using the information. Robotic control architectures define how these abilities should be integrated to construct and develop an autonomous navigation. The control architectures could be classified into three categories: Deliberative (Centralized) navigation, Reactive (Behaviour-based) navigation and hybrid (Deliberative - Reactive) navigation. This paper reviews various control architectures for autonomous navigation of mobile robots. The significance, advantages and drawbacks of the architectures are discussed and compared with each other.

Key words: Intelligent system, mobile robot, autonomous navigation, control architecture.

INTRODUCTION

Since early 1960’s, researches on mobile robot navigation have been gradually increased. Different applications for mobile robots represent different navigation problem. It is essential for intelligent mobile robots to sense and perceive the working environment, plan a path, make a decision and execute appropriate reaction using the information (Vuković and Miljković, 2009). Control architectures define how these abilities should be integrated to get desired results. Therefore, various control architectures have been proposed to design and develop of robust, flexible, reliable and high-performance control systems for autonomous navigation of mobile robot. Each of the control architectures implies new concepts and solutions to deal with the navigation problem. The control architectures could be classified into three categories: Deliberative (Centralized) navigation, Reactive (Behaviour-based) navigation and hybrid (Deliberative - Reactive) navigation. The deliberative (Centralized) navigation architecture (Chatila and Laumond, 1985; Giralt et al., 1984; Laird and Rosenbloom, 1990; Moravec and Cho, 1989; Thorpe, 1993; Jochem et al., 1995) creates a global model of working environment through sensory system or user input.

Deliberative planning searches for the optimal path and generates a plan to reach the goal. Then the execution system is applied to perform an action within the context of the static model of environment and planning system to accomplish a given task (Huq et al., 2008). The reactive (Behaviour-based) architectures (Ye and Wang, 2000; Kasper et al., 2001; Seraji and Howard, 2002; Yang et al., 2006; Motlagh et al., 2008) were developed from 1980’s to deal with shortcomings of the deliberative approaches in dynamic and unknown environments. These architectures generate control commands based on currently perceived environment. Therefore, it is not necessary to build a complete model of environment and the sensed data directly couples to the robot’s actuators using a particular set of transfer functions called task-achieving modules or behaviours.

Hybrid control architecture first was developed by Arkin (1989, 1998) and Murphy (2000) which involves the advantages of planning in deliberative architectures and quick response of reactive architectures in dynamic or unknown environment. Then, the hybrid control architectures (Weitzenfeld and Arkin, 1998; Averbukh,
Deliberative control architecture

The Deliberative (Centralized) navigation architecture is the oldest schema in artificial intelligence (AI). The deliberative techniques (Schwartz and Sharir, 1983; Nagatani et al., 1998; Canny, 1987; Mitchell, 1986; Takahashi, 1989; Latombe, 1991; Pruski and Rohmer, 1997) use a global world model provided by user input or sensory information to generate appropriate actions for the mobile robot to reach the target. As shown in Figure 1, the deliberative control architecture comprises three modules: sensing, planning and action modules. First robot sense it’s surrounding and creates a world model of static environment by combining sensory information. Then it employs planning module to search an optimal path toward the goal and generate a plan for robot to follow.

Finally, robot executes the desired actions to reach the target. After a successful action, robot stops and updates information to perform the next motion. Then, it repeats the process until it reaches the goal (Huq et al., 2008; Yang et al., 2006). Top-down approach in planning module is an important characteristic of this architecture where high level constraints are broken into low level commands. It can coordinate multiple goals and constraints within a complex environment (Huq, 2008). However, in deliberative navigation, accurate model of environment is needed to plan a globally feasible path. It is difficult to obtain a completely known map. To perform necessary calculations, enormous processing capabilities and memory is needed. Moreover, the top-down approach of planning produces delay in navigation process and if any modules do not function properly, the system may fail entirely.

Therefore, these approaches are not proper in the presence of uncertainty in dynamic or real world.

Reactive control architecture

Reactive (behaviour-based) navigation architecture was developed by Brooks (1986) to tackle the deliberative navigation problems in dynamic and unknown environments. These approaches generate control commands based on current sensory information. To take actions, the robot uses the local model of environment without planning process. Therefore, it is not necessary to build a complete model of environment. Bottom-up approach for decision making is used in the behaviour based architectures in which high level constraints are not integrated in action generation process.

Reactive navigation has a quick response in the dynamic and unknown environment. Figure 2 represents the overall architecture of behaviour-based approaches. In first layer, robot gathers sensory information. Then a transfer function called behaviour receives particular sensory inputs perception and transforms them into the predefined response. Finally, the robot executes an action based on the output of active behaviour. In fact, complex navigation tasks are broken down into several simpler and smaller sub-level tasks which improve the total performance of the navigation system. Two basic behavior-based control architectures include subsumption architecture (Brooks, 1986) and motor schemas (Arkin, 1989). Subsumption control architecture was introduced by brooks (1986) at Massachusetts Institute of Technology (MIT).

The subsumption behavior-based control system is
composed of several layers of task-achieving behaviors where each behavior can receive sensory information for a given task (obstacle avoidance, wall following, target seeking, etc.). The task achieving process increases responsiveness and reduces planning complexity of the control system to a dynamic and unknown environment. In subsumption architecture, the planning module is eliminated from the control architecture and the focus is exclusively on the sensing and acting modules. Unlike the deliberative approaches, the behaviors provide a direct coupling between sensory inputs and robot’s actions. As Figure 3 shows, in subsumption architecture, behaviors are layered and each behavior receives particular sensory information.

Coordination of behavior layers refers to the priority-based arbitration. Priority-based arbitration is a process of deciding which behavior to be active when multiple conflicting behaviors are trigged (Dupre, 2007). Therefore, the highest active behavioral module generates the overall output of architecture. Another behavior-based architecture proposed by Arkin (1998) is motor schemas architecture (Figure 4) which was motivated by biological sciences based on perceptual schemas. The motor schemas theory explains the motor behavior in terms of the simultaneous control of many different activities. Each behavior can produce an output in the vector form. These outputs are combined and then the overall response of the system is achieved by the vector summation. For example, a potential field can be defined as the output of each schema. The commanded movement is a vector which is the superposition of all fields.

Subsumption architecture advocates the competitive selection of behaviours, while the motor schemas rely on the use of cooperative coordination. Motor schema provides an ability to simultaneously use the outputs of more than one behavior with capturing their particular influence on overall output (Vuković and Miličković, 2009).

The overall advantages of behaviour-based navigation systems are:

i) Their ability to build a navigation system in an incremental way of layer upon layer.

ii) Their quick reaction to the unknown and dynamic environment.

iii) They do not require modelling and storing the whole model of the environment.

iv) There is less computation and shorter delay between perception and action.

v) And they are more robust and reliable which means in case of a behaviour unit failure, the other units continue the tasks.

The drawbacks of behaviour-based control are as follows:

i) Difficulty in coordination among the behaviours, the interaction between the system and environment is difficult and less predictable.

ii) Behaviours are low level so they do not reflect high level tasks.

iii) Lack of planning module could be not appropriate for some complicated tasks.

Hybrid control architecture

Although reactive navigation architectures established a successful framework for mobile robot navigation, there are still some problems in regards with the complex unknown environment. To perform an autonomously navigation in a real world some features of deliberative architecture combines with the reactive architecture which called hybrid architecture. The hybrid control architecture could be classified to three styles (Murphy, 2000): Managerial style, state hierarchies and model-oriented. In Managerial Style (Yavuz and Bradshaw, 2002; Arkin, 1987; Busquets et al., 2003; Kolp et al., 2006; Kim and Chung, 2006), the deliberative module is in charge of planning in higher level. Then, the plans are sent to low level which is reactive module to be implemented. Each module attempts to modify the problems and solve them itself or refines by superior module if cannot solve its own problem.

The State-hierarchies (Bonasso et al., 1997; Lindstrom, 2000) use the knowledge of robot’s state in past, present
and future. The deliberative layer requires the robot’s past state (what the robot has done) to predict the future (path planning). The reactive layer functions in the present state (self-awareness) to complete deliberative planning instruction and generate the robot’s motion. A model-oriented style (Konolige, 1997) more concentrates on global model of environment and it is similar to the deliberative architecture. However, the updated global model of environment is used by reactive layer immediately to reduce deliberative processing time (Davies, 2007).

The common hybrid control architectures consist of three layers: deliberative layer, control execution layer and reactive (behavior-based) layer (Figure 5). The deliberative navigation is applied for high level issues to develop an optimal plan. The high level constraints consist of sensor fusion, map building and planning. Then, the optimum commands from the higher level are sent to the reactive layer to generate the robot’s action. The execution layer (behaviour-coordinator) is responsible to supervise the interaction between the high level layer and low level layer (Perez, 2003). The integration of various features in the hybrid architecture form a novel flexible and robust architecture for control systems. The hybrid control architecture specifications in comparison with the deliberative and reactive architectures are summarised in Table 1. Next, each specification is described in Table 2.

**Conclusion**

Various control architectures for autonomous navigation of mobile robot have been described and compared in this paper. Among the proposed architectures, the

---

**Table 1. Analysis of the control systems architecture.**

<table>
<thead>
<tr>
<th>Architecture Specification</th>
<th>Deliberative</th>
<th>Reactive</th>
<th>Hybrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal oriented</td>
<td>Very good</td>
<td>Not good</td>
<td>Good</td>
</tr>
<tr>
<td>Flexibility</td>
<td>Very bad</td>
<td>Very good</td>
<td>Very good</td>
</tr>
<tr>
<td>Ease of application</td>
<td>Very bad</td>
<td>Very good</td>
<td>Good</td>
</tr>
<tr>
<td>Reactivity</td>
<td>Very bad</td>
<td>Very good</td>
<td>Good</td>
</tr>
<tr>
<td>Optimal operation</td>
<td>Very good</td>
<td>Very bad</td>
<td>Good</td>
</tr>
<tr>
<td>Task learning</td>
<td>Very good</td>
<td>Moderate</td>
<td>Moderate</td>
</tr>
<tr>
<td>Robustness</td>
<td>Not good</td>
<td>Good</td>
<td>Very good</td>
</tr>
<tr>
<td>Planning</td>
<td>Very good</td>
<td>Not good</td>
<td>Good</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Not good</td>
<td>Very good</td>
<td>Very good</td>
</tr>
</tbody>
</table>

---

**Figure 5. Hybrid control architecture (Perez, 2003).**
Table 2. Description of the specifications used in the evaluation of the control architectures.

<table>
<thead>
<tr>
<th>Specification</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal oriented</td>
<td>Capability of the control system to provide means to accomplish multiple goals</td>
</tr>
<tr>
<td>Flexibility</td>
<td>Ability of adding new sub-systems or making any modifications and additions to a system functions without disrupting the established functionality (Yavuz and Bradshaw, 2002)</td>
</tr>
<tr>
<td>Ease of application</td>
<td>Refers to ease of an architecture to be understood, developed, tested and debugged</td>
</tr>
<tr>
<td>Reactivity</td>
<td>Ability of a system to respond and adapt to the sudden changes in the environment</td>
</tr>
<tr>
<td>Optimal operation</td>
<td>Capability of a system to obtain optimal cost function in motion criteria such as distance, time, oscillation, etc.</td>
</tr>
<tr>
<td>Task learning</td>
<td>Ability of the system to learn through a teach mode or operation to carry out specific tasks</td>
</tr>
<tr>
<td>Robustness</td>
<td>Capability of a system to handle sudden changes, imperfect inputs, and unexpected malfunctions</td>
</tr>
<tr>
<td>Planning</td>
<td>A set of partially ordered tasks for the robot to perform and work on a problem at the highest level of abstraction possible so as to make its problem space as small as possible until a plan is finished</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Contains the capabilities and performance of a system to maximize individual utility and cooperation of subtasks to generate an optimized and smooth trajectory</td>
</tr>
</tbody>
</table>

deliberative architecture was more promising in high level control to obtain optimal path toward the target. However, it fails in dynamic or unknown environments. The reactive architectures had a better performance dealing with uncertainties for fast obstacle avoidance of the robot in unknown or dynamic environment but still have some difficulties dealing with complicated tasks.

To achieve a comprehensive navigation, robot needs more abilities that exceeds deliberative and reactive paradigms such as perception and world representation ability to enable information gathering and processing, fast reacting for static or dynamic obstacle avoidance, map building ability to insure the robot to be able to localize itself relative to the environment, inference and decision making ability to make reliable decisions based on that particular information. Therefore, the deliberative and reactive architectures have been combined and formed hybrid control architecture in the way to cope with the navigation problems. The review of the various control architectures showed that the hybrid scheme has the best performing supervisory control architecture and it is more prosperous and promising dealing with unknown, dynamic navigation problem.

REFERENCES


Dupre ME (2007). GA optimized fuzzy control of an autonomous mobile robot. Faculty of Graduate Studies, University of Guelph.


