Behavioural Architecture for a Differential-Drive Mobile Robot

Carlos Delgado-Mata, Ramiro Velázquez, and Carlos A. Gutiérrez

IPIT (Instituto Panamericano de Investigación Tecnológica),
Escuela de Ingenieria, Universidad Panamericana campus Bonaterra,
Aguascalientes, México, C.P. 20290 cdelgado@up.edu.mx

Abstract. This paper presents the work in progress of an ethologically inspired Action Selection Mechanism to control a Differential-Drive Mobile Robot. The mathematical model of a two wheel differential-drive model is presented. The model shows how zero turning radius is achieved with only bidirectional movement. Behaviour patterns are used to map the incoming stimuli from ultrasound sensors into responses that affect the voltage's intensity of each wheel's motor. Therefore, it performs translational and rotational movements of the mobile robot described herewith.

1 Introduction

Understanding how wheeled mobile robots (WMR) move in response to input commands is essential for feedback control design and many navigation tasks such as path planning, guidance, and obstacle avoidance.

Campion and Chung classified in [1] the mobility of WMR into five generic structures corresponding to a pair of indices (m, s): mobility degree m and steerability degree s. The first one refers to the number of degrees of freedom the WMR could have instantaneously from its current position without steering any of its wheels while the second refers to the number of steering wheels that can be oriented independently in order to steer the WMR.

1.1 Action Selection Mechanisms

Action Selection has been quintessential in fields like simulation of adaptive behaviour. Applications are constantly developed in Robotics, Intelligent Agents, and lately in Virtual Agents populating Virtual Environments, that is embodied virtual agents. Historically, there have been two approaches for selecting actions: the reactive [2] and the deliberative [3]. The advantages of the former [4] is that they are computationally cheap, and can adapt better to a changing environment. The advantages of the latter is that they can hold in memory a representation of the world and thus they -in theory- could accomplish a more informed and better solution than their counterparts. They do not suffer from the local minima problems inherent in local decision making. Motivated behaviours are governed not only by environmental stimuli but also by the internal state of the animal, being influenced by such things as appetite.

Advances in Computer Science and Electronic Systems
Research in Computing Science 52, 2011, pp. 355-364
Received 23/01/11
Accepted 20/02/11
Final version 07/03/11

The rest of the paper is organized as follows: in Section 2, the popular two wheel differential-drive model is obtained using the general two-active-fixed wheels and one-passive-caster wheel structure. In Section 3 a Behavioural Architecture to drive the two wheel diferential-drive robot is described. Finally, the conclusion summarizes the paper main concepts.

2 Wheeled Mobile Robot

The five mobility WMR classes classified by Campion [1], that correspond to a pair of indices (*m*, *s*): mobility degree *m* and steerability degree *s*, are:

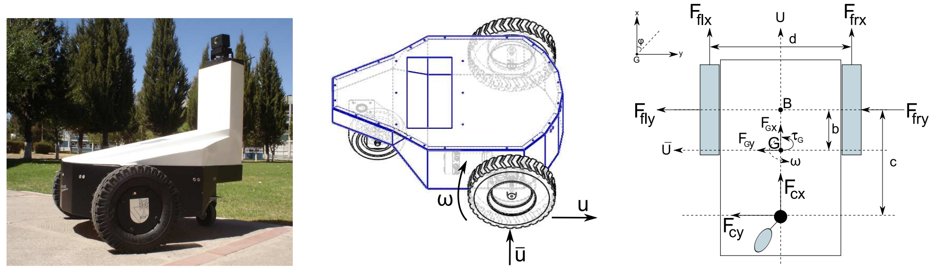
- Type (3,0) robots or omnidirectional robots have no steering wheels (s=0) and are equipped only with Swedish or caster wheels. They have full mobility in the plane (m=3), which means that they are able to move in any direction without any reorientation.
- Type (2,0) robots have no steering wheels (s=0) but either one or several fixed wheels with a common axle. The common axle restricts mobility to a two-dimensional plane (m=2).
- Type (2,1) robots have no fixed wheels and at least one steering wheel. If there is more than one steering wheel, their orientations must be coordinated (s=1). Therefore, mobility is restricted to a two-dimensional plane (m=2).
- Type (1,1) robots have one or several fixed wheels on a common axle and also one or several steering wheels, with two conditions for the steering wheels: their centers must not be located on the common axle of the fixed wheels and their orientations must be coordinated (s=1). Mobility is restricted to a one-dimensional plane determined by the orientation angle of the steering wheel (m=1).
- Type (1,2) robots have no fixed wheels, but at least two steering wheels. If there are more than two steering wheels, then their orientation must be coordinated in two groups (s=2). Mobility is restricted to a one-dimensional plane (m=1) determined by the orientation angles of the two steering wheels.

This paper particularly address type (2,0) robots.

2.1 Differential-Drive WMR

The wheeled mobile robot described herein is a type (2,0) robot. There are many design alternatives; however, the two-wheel differential-drive robot is by far the most popular design.

Let us consider our prototype IVWAN (Fig. 1(a)). Its mechanical structure is based on a differential-drive configuration consisting of two independently controlled front-active wheels and one-rear-caster wheel (Fig. 1(b)). Active wheels are driven by two high-power DC motors which allow IVWAN to achieve a maximum speed of 20 km/hr. IVWAN exhibits both manual and autonomous operation: it can be tele-operated or self-guided by a color camera and an array of ultrasonic sensors that allow the machine to detect and follow visual patterns and negotiate obstacles, respectively [5].



(a) (b) (c)

Fig. 1. Type (2,0) WMR IVWAN (Intelligent Vehicle With Autonomous Navigation): (a) prototype and (b) its differential-drive structure. Two front wheels each driven by its own motor. A third wheel is placed in the rear to passively roll along while preventing the robot from falling over. The wheels exhibit three speeds: u, $\bar{u}$, and ω. (c) Free-body diagram. The first subscript stands for front **f** and caster **c** wheel while the second subscript stands for right **r** and left **l** wheel.

Fig. 1(c) shows a schematic representation of the differential-drive structure. Here, **B** represents the center of the axis connecting both traction wheels; **G** represents the vehicle's center of mass and for simplicity, it is considered as the point to control in position (**x**, **y**) and orientation (φ).

Resultant forces and momentum in the structure can be expressed by eq. (1):

$$\begin{aligned}
\sum F_x &= m(\dot{u} - \bar{u}\omega) = F_{frx} + F_{flx} + F_{cx} + F_{Gx} \\
\sum F_y &= m(\dot{\bar{u}} + u\omega) = F_{fry} + F_{fly} + F_{cy} + F_{Gy} \\
\sum M_z &= I\dot{\omega} = \frac{d}{2}(F_{frx} - F_{flx}) - b(F_{fry} + F_{fly}) + \\
&\quad + (c-b)F_{cy} + \tau_G
\end{aligned} \tag{1}$$

where **m** is the vehicle's total mass, **I** is the moment of inertia around point **G**, and **u**, $\bar{u}$ and ω are the robot's linear, transverse sliding, and angular speeds, respectively (Fig. 1(b)). Speed $\bar{u}$ can be reasonable neglected assuming that the wheels do not slip during motion. Concerning **u** and ω, they can further be defined by eq. (2):

$$\begin{aligned}
u &= \frac{1}{2}[r(\omega_r + \omega_l) + (u_r + u_l)] \\
\omega &= \frac{1}{d}[r(\omega_r - \omega_l) + (u_r - u_l)]
\end{aligned} \tag{2}$$

where **r** is the traction wheel radius, **d** is the distance between the traction wheels (see Fig. 1(c)), ω_r, and ω_l are the angular speeds of the right and left wheels respectively, and $\mathbf{u}_r$ and $\mathbf{u}_l$ are the linear speeds of the right and left wheels respectively.

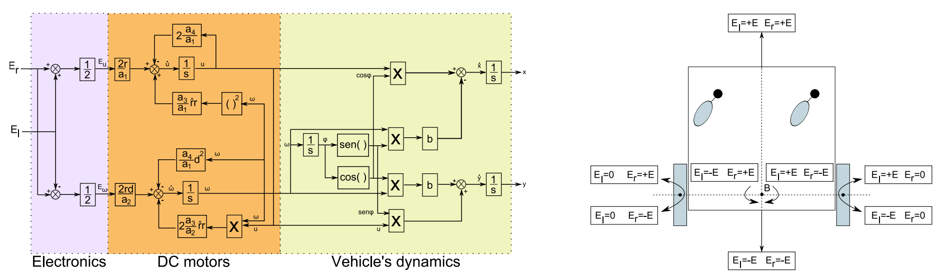
Kinematics of point **G** is related to **u** and ω by eq. (3):

$$\begin{aligned}
\dot{x} &= ucos\varphi - b\omega sin\varphi \\
\dot{y} &= usin\varphi + b\omega cos\varphi \\
\dot{\varphi} &= \omega
\end{aligned} \tag{3}$$

As aforementioned, traction wheels are powered by DC motors. These can be modeled by eq. (4):

$$\begin{aligned} \tau_r &= \frac{k_a}{R_a}(E_r - k_b\omega_r) \\ \tau_l &= \frac{k_a}{R_a}(E_l - k_b\omega_l) \end{aligned} \tag{4}$$

where τ_r and τ_r are the torques developed by the motors on the right and left wheels upon input DC voltages $\mathbf{E}_r$ and $\mathbf{E}_l$ respectively, $\mathbf{k}_a$ and $\mathbf{k}_b$ are the motor's torque and electromotive force constants, and $\mathbf{R}_a$ is the motor's electric resistance. Inductive voltages have been neglected.



(a) (b)

Fig. 2. (a) Block diagram reference for differential-drive robots. (b) Summary of motion upon voltages $\mathbf{E}_r$ and $\mathbf{E}_l$.

Equations describing the wheel-motor system can be simply written as shown in eq. (5):

$$\begin{aligned} I_e\dot{\omega}_r + D_e\omega_r &= \tau_r - F_{frx}\hat{r} \\ I_e\dot{\omega}_l + D_e\omega_l &= \tau_l - F_{flx}\hat{r} \end{aligned} \tag{5}$$

where $\mathbf{I}_e$ and $\mathbf{D}_e$ are the moment of inertia and the coefficient of viscous friction of the wheel-motor system, respectively and $\hat{r}$ is the nominal radius of the traction wheel tires. Using and combining eqs. (1) to (5), the differential-drive model can be summarized by eq. (6):

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \\ \dot{u} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} ucos\varphi - b\omega sin\varphi \\ usin\varphi + b\omega cos\varphi \\ \omega \\ \frac{a_3}{a_1}\hat{r}r\omega^2 - 2\frac{a_4}{a_1}u \\ -2\frac{a_3}{a_2}\hat{r}ru\omega - \frac{a_4}{a_2}d^2\omega \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{2r}{a_1} & 0 \\ 0 & \frac{2rd}{a_2} \end{bmatrix} \begin{bmatrix} E_u \\ E_\omega \end{bmatrix} \tag{6}$$

with inputs:

$$E_u = \frac{E_r + E_l}{2}$$
$$E_\omega = \frac{E_r - E_l}{2}$$

and constants:

$$a_1 = \frac{R_a}{k_a}(m\hat{r}r + 2I_e) \qquad [V \cdot s^2]$$
$$a_2 = \frac{R_a}{k_a}[I_e d^2 + 2\hat{r}r(I + mb^2)] \qquad [V \cdot m^2 \cdot s^2]$$
$$a_3 = \frac{R_a}{k_a}mb \qquad [V \cdot s^2/m]$$
$$a_4 = \frac{R_a}{k_a}(\frac{k_a k_b}{R_a} + D_e) \qquad [V \cdot s/rad]$$

Note that eq. (6) relates the robot's motion to the motors' input voltages. The block diagram model for differential-drive robots is shown in fig. 2(a). This diagram identifies the electronics, DC motors, and the vehicle's dynamics.

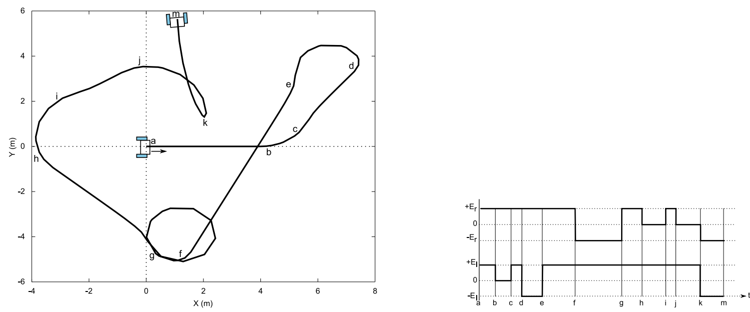
Fig. 2(b) summarizes how differential-drive robots are controlled by the input voltages $\mathbf{E}_r$ and $\mathbf{E}_l$. When both voltages are equal, the two driving wheels turn at the same angular speed and in the same direction, which causes a translation movement. If one voltage is set to zero, one of the wheels turns while the other remains motionless, then the robot describes a circle centered on the motionless wheel. If both voltages are equal in magnitude but opposite sign, the wheels turn at the same speed but in opposite direction which causes a rotation around the center of the axis connecting both wheels (point **B**). Note a zero turning radius in this case.

Numerical values of the parameters involved in eq. (6) can be easily measured from an existent prototype and the specifications of the DC motors can be obtained from the manufacturer. As illustrative example, consider all gain blocks of fig. 2(a) as unity gains. Fig. 3(a) shows a computer simulation of a certain trajectory in the XY plane. Fig. 3(b) shows the driving signals supplied to the DC motors. Note the correspondence with fig. 2(b).

The next section presents the Action Selection Mechanism used to control the differential-drive mobile robot described herewith.

3 Behavioural architecture

This section presents the behavioural architecture used to drive the WMR. The architecture was originally developed for multiple cooperating robots - the Behavioural Synthesis Architecture or BSA [6] - and reapplied it to agents in a virtual environment (VE) in the Virtual Teletubbies project [7]. An object oriented approach (BAMUVA) was developed to simulate conspecific virtual mammals; it is described in [8]. The BSA incorporated three structures at increasing levels of abstraction: behaviour patterns, behaviour packets, and behaviour scripts. An overview of this architecture is presented henceforth.



(a) (b)

Fig. 3. (a) A simulated trajectory of the differential-drive robot and (b) the corresponding driving signals.

3.1 Behaviour patterns

At the most primitive level, a behaviour pattern, (**bp**), was defined as a pair of functional mappings, one from incoming sensory stimulus to outgoing desired motor response, and the other from incoming sensory stimulus to utility. That is, a mapping to define the importance of the motor response for the given level of stimulus. A mobile robot, like the one described herein, possesses a repertoire of behaviour patterns, with each active pattern at any given time proposing its desired motor response (voltage intensity for each high-power DC motor) according to its current sensory input (sonar sensors). These responses are weighted by their utility values and synthesised together to produce an emergent response; the actual behaviour of the mobile robot. Thus, second-to-second variation in emergent behaviour was dealt via weighted synthesis on a continuous basis, unlike the time-sliced Brooksian architecture [4].

The basic component in the architecture is the behaviour pattern, **bp**, where

$$\mathbf{bp} = \begin{bmatrix} r \\ u \end{bmatrix} \tag{7}$$

and

$$r = f_r(s) \tag{8}$$

$$u = f_u(s) \tag{9}$$

r is the desired motion response and is a function, f_r, of a given sensory stimulus, s.

Associated with every response is a measure of its utility or importance, u. This quantity is a function , f_u, of the same sensory stimulus. Hence a **bp** defines not only what the motion response should be for a given sensor input, but it also provides a measure as how the relative importance of this response varies with respect to the same sensor input. The values of r and u constitute a vector known as *utilitor*. Figure 4 shows an example of a simple **bp** that might exist at a given level. Consider the situation where the sensory stimulus relates to a mobile robot's forward facing distance to an obstacle measuring sensor and the associated motion response relates to the forward velocity for the mobile robot. From figure 4 it can be seen that as the mobile robot gets nearer to the object, its forward translate velocity will be reduced to zero. At the same time the associated utility for the motion response increases. Thus, as the mobile robot gets nearer to an object in its path, the more important it becomes to the mobile robot to slow down. At any point in time, t, multiple conflicting motion responses are typically generated. For example, a mobile robot may be moving towards a goal location when an obstacle unexpectedly appears in its path and at the same time senses that it *needs* to replenish its battery. In such situation what should it do? In BAWMR (Behaviour Architecture for a Wheeled Mobile Robot), conflicting motion responses are resolved by a behaviour synthesis mechanism to produce a resultant motion response. Competing utilitors are resolved by a process of linear superposition which generates a resultant utilitor, UX_t where:

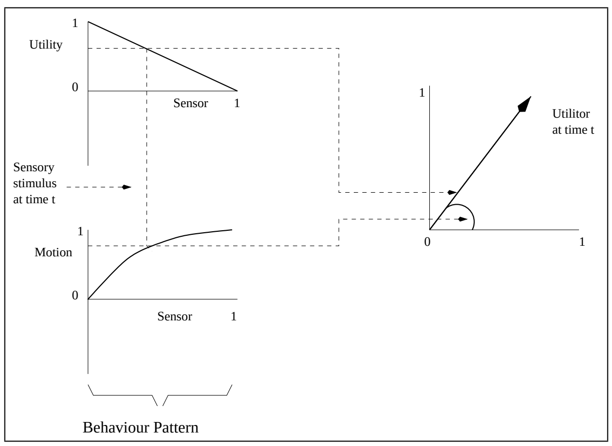


Fig. 4. Behaviour pattern example

$$\mathbf{UX_t} = \sum_{n=1}^{m} u(t,n) \cdot e^{j \cdot r(t,n)} \tag{10}$$

and m equals the total number of related utilitors generated from the different behaviour levels concerned with motion. Given a resultant utilitor, a resultant utility, uX_t, and a resultant motion response, rX_t are simply obtained from

$$\mathbf{yX_t} = \frac{|U\ X_t|}{m} \tag{11}$$

$$\mathbf{yX_t} = arg(U\ X_t) \tag{12}$$

X identifies the relevant degree of freedom, e.g. forward movement, and the result motion response, rX_t, is then executed by the mobile robot. From equation 4, it can be seen that generating a resultant utilitor from different behaviours within the architecture constitutes a process of additive synthesis, as shown in figure 5

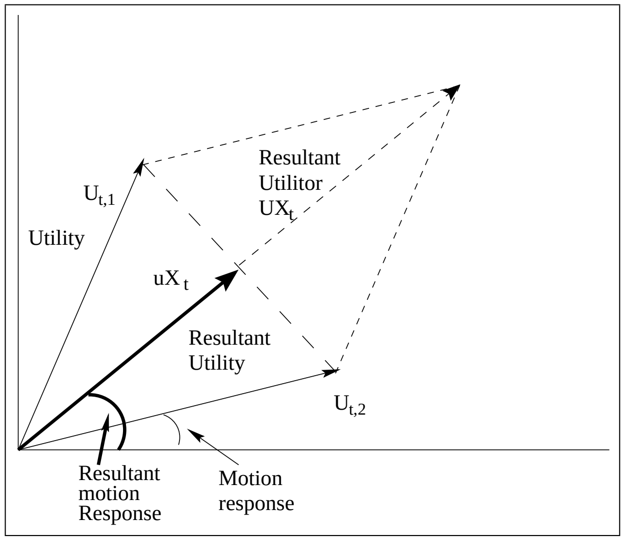


Fig. 5. Generating a resultant utility and motion response from two constituent utilitors

The BAWMR architecture is an Object Oriented extension to the Behaviour Synthesis Architecture, which was developed at the University of Salford [9], to accomplish a task through cooperating robots[6]. This work used ethological knowledge similar to the one described in Animal Behaviour literature [10]. While each robot had a repertoire of simple behaviour patterns, complexity emerged through interactions between behaviour patterns and between robots.

3.2 Behaviour packets

If all the **bp**s in an agent's repertoire were active at the same time then the overall emergent behaviour of the agent might be of little value. For example, patterns designed to produce obstacle avoidance (as described above) are not useful if you want an agent wait for a particular stimuli. The **bp** designer must always bear in mind that the low-level architecture is sensor-driven, and not task or even sub-task dependent. What is needed in this case is an automatic mechanism for deactivating the 'obstacle avoidance' **bp**s when the 'waiting' **bp**s is active. Associated therefore with every **bp** within the mobile robot is an 'active flag', which enables or disables it. Thus obstacle avoidance **bp**s for example can be turned off and on when required. A **bp** is 'deactivated' in the BSA by forcing the respective utility to zero. The action effectively produces a **bp** of zero importance and hence one which does not contribute to the overall emergent behaviour of the agent.

This mechanism is applied by grouping together **bp**s in goal-achieving sets known as behaviour packets. A behaviour packet is a small data structure which includes a sensory pre-condition for activating the **bp**s it references, and a sensory post-condition which controls deactivation of the named **bp**s. Behaviour packets show some similarity with AI production rules [11], though they work at the sub-symbolic level and are driven by incoming sensor data rather than by an inferencing system. They support behavioural sequencing for agents performing at a task (universe) behaviour level. Thus, a sensory pre-condition of 'being near the goal' could be used to move from a behaviour packet in which obstacle avoidance **bp**s were active to one in which they are not.

Therefore, behaviour packets provide a mechanism for contextually sensitive behaviour switching, which is seen as a more flexible mechanism than the finite-state machine definition of inhibition and excitation between behaviours of the subsumption architecture [4].

3.3 Behaviour Script: high-level sequencing and agent drives

A behaviour script is simply a set of behaviour packets assembled for the achievement of a particular task, using the sensory pre-and post-conditions. The original approach was to generate behaviour scripts on the fly using a reflective agent incorporating a symbolic AI planner, and then send the individual scripts to behavioural-based agents. This hybrid approach was taken with the co-operative robots in [12] and is appropriate where the domain is predominantly task-based.

The default script executes a single packet containing **bp**s that effectively lets the low-level module handle wandering in the environment while avoiding obstacles. The default script is changed when another sensory precondition from another set of packets is met.

4 Conclusion

This paper intends to present simple and reliable mathematical model for different design of a type (2,0) robot. In particular, this draft has presented the differential-drive model: the general two-active-fixed wheels and one-passive-caster wheel as well as the belt-drive system. An Action Selection Mechanism to drive the mobile robot was presented. The stimulus received from the sonar sensors is mapped via the BAWMR (Behaviour Architecture for a Wheeled Mobile Robot) into motor responses that affect the voltage's intensity of each of the high-power DC motors. This particular mobile robot is suitable for a non-static environment, which are the most common.

References

1. G. Campion and W. Chung, *Springer Handbook of Robotics*, ch. Wheeled robots, pp. 391–410. Berlin Heidelberg: Springer Verlag, 2008.
2. T. Tyrrell, *Computational Mechanisms for Action Selection*. PhD thesis, University of Edinburgh, Edinburgh, Scotland, 1993.
3. N. Nilsson, *Artificial Intelligence: A New Synthesis*. Morgan Kauffman, 1998.

4. R. A. Brooks, “A robust layered control system for a mobile robot,” *IEEE Journal of Robotics and Automation*, vol. 2, no. 1, pp. 14–23, 1986.
5. J. Martinez, G. Moran, B. Romero, A. Camacho, D. Gutheim, J. Varona, and R. Velazquez, “Multifunction all-terrain mobile robot ivwan: design and first prototype,” in *Proc. of 2nd Israeli Conference on Robotics*, (Herzlia, Israel), 2008.
6. D. P. Barnes, R. A. Ghanea-Hercock, R. S. Aylett, and A. M. Coddington, “Many hands make light work? an investigation into behaviourally controlled cooperant autonomous mobile robots,” in *Proceedings of the first International Conference on Autonomous Agents* (L. Johnson, ed.), (Marina del-Rey, Calif.), pp. 413–20, ACM Press, 1997.
7. R. S. Aylett, A. Horrobin, J. J. O’Hare, A. Osman, and M. Polyak, “Virtual teletubbies: reapplying a robot architecture to virtual agents,” in *3rd International Conference on Autonomous Agents*, (Seattle Washington), 1999.
8. C. Delgado-Mata and R. Aylett, “Communicating emotion in virtual environments through artificial scents,” in *LNCS Intelligent Virtual Agents* (E. de Antonio and R. Aylett, eds.), (Madrid, Spain), Springer-Verlag, 2001.
9. D. P. Barnes, “A behaviour synthesis architecture for cooperant mobile robot,” in *Advanced Robotics and Intelligent Machines* (J. O. Gray and D. G. Caldwell, eds.), pp. 295 – 314, United Kingdom: IEE Control Engineering Series 51, 1996.
10. D. McFarland, *Animal Behaviour*. England: Addison Wesley Longman, 1999.
11. R. Davis and J. King, *An overview of Production Systems in Machine Intelligence*, pp. 300–332. Wiley and Sons, 1977.
12. R. S. Aylett, “Communicating goals to behavioural agents - a hybrid approach,” in *Proceedings of the Third World Congress on Expert Systems*, (Seoul), 1996.