A brief comparison between the Subsumption Architecture and Motor Schema Theory in light of Autonomous Exploration by Behavior Based Robots

Dip N. Ray[1*], S. Mukhopadhyay[2], and S. Majumder[1]
[1] Surface Robotics Lab, Central Mechanical Engineering Research Institute (CSIR), Durgapur, India
[2] Mechanical Engineering Department, National Institute of Technology, Durgapur, India

[*] Corresponding author (email: dnray@cmeri.res.in)

Abstract

Research and development of mobile robot can broadly be divided into two major paradigms, viz. Sense → Plan → Act (SPA) and Behavior Based. Although, majority of the mobile robots developed and reported in the literature follows the SPA approach; yet behavior based research is gaining momentum in the recent years. Reactive architecture is commonly used for implementing behaviors. In general, behavior based systems operate through a set of multiple independent tasks for achieving goals [Grey Walter, Brooks, Arkin, Maja J. Mataric, Mahadevan, Dorigo etc.]. This however calls for modular and layered system architecture [Brooks, 1991]. Subsequently, Motor Schema theory [Arkin, 1989] has been proposed and successfully applied in a few experimental systems. The work reported in this paper compares these two approaches through experimental mobile robots developed at CMERI. The major difference between these two approaches can be summarized as: Subsumption architecture is essentially a priority based approach whereas the Motor Schema theory leads to division of task into multiple smaller and simpler tasks. In this sense "Motor Schema" is an obvious successor of "Subsumption architecture" originally proposed by Brooks.

The Autonomous Robot Based on Intelligent Behaviors (ARBIB-I) uses classical subsumption architecture. The tasks consist of a set of parallel activities encapsulating the sensing, perception & control system. Each of which is assigned with fixed priority values. Various sensors used, such as Infra Red sensors, Battery level detectors, sound sensor and any other measure the system response and were assigned with priority depending upon the tasks envisaged. The obstacle avoidance behavior has been assigned with higher priority over other task, such as wandering, escaping for example. On the other hand, ARBIB-II is an improved design of ARBIB-II, uses Motor Schema theory for its operation. Here the task "Reach Goal" is divided into different behaviors (schemas) such as Move-ahead, Avoid-static-obstacles, Avoid-moving-obstacles, Move-to-goal or any other well perceived behavior. The action selection behavior selects the right motor schema suitable for the goal. Performance of these two systems is compared for autonomous exploration within a constrained laboratory space. The data generated from these two experiments will be used for implementing learning behavior for which work is in progress.

Keywords: Motor Schema, Subsumption architecture, Behavior based exploration

1. Introduction

Behavior Based Robotics (BBR) is related to behavior-oriented AI. It is different from the conventional or classical AI in the sense that BBR does not plan beforehand, like the classical AI. It is basically based on different behaviors, like 'aggression', 'fear', 'hunger', 'avoid', 'turn', 'move' etc., and numbers of behaviors are running in parallel to execute a task. 'Behavior' is referred to the transfer of environmental stimulus to a particular type of response as shown in Fig–2. The transfer is very simple and doesn't involve complex computation. In that sense it is having real intelligence as it decides itself what to do autonomously. Rodney Brooks [1] [2] spoke out about the concept of Behavior Based Robotics in 80's. But before Rodney Brooks, in 50's, Grey Walter had made a light sensing tortoise (Elsie or Grey Walter's

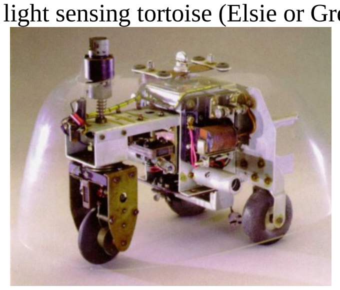

Fig-1: The first announced photo of Grey Walter's Tortoise (Courtesy: Burden Neurological Institute)

Tortoise) using phototubes and photocells that was capable of moving towards light source when the system needed to be charged (this behavior of light was known as ‘aggression’) and moved away from the light when the system became fully charged (‘fear’).

2. Classical Robotics and Behavior based robotics

Research is going on BBR all over the world. A comparative study described in the following paragraphs will help to understand the concept, its scope and prospects.

In robotics, control is a big issue for obtaining intelligence, collision avoidance, path planning and autonomous navigation [3]. The existing conventional/classical robotics has some control mechanism, which guides the end effectors to act accordingly, after it analyzes the inputs, obtained from various sensors and sends responses to those end effectors. The inputs from various sensors are used intermediately to symbolically represent the environment or action. On the basis of this environmental model planning is done and then only commands are sent to the end effecters. But if the end effecters are directly coupled to those sensors and there is an intelligent agent to control the system individually, then it will be able to take decision itself. So, this is one kind of intelligence, we often search in robots. This behavior is often called ‘Reactive’ in nature like the closing of eyes due to intense light in human beings. The nerves associated with the eyelids close the eyes in no time without receiving any signal from the brain. This is a spontaneous process. As the decision is taken locally the response time is very less and the control is decentralized in nature.

STIMULUS $\longrightarrow$ $\boxed{\text{BEHAVIOR}}$ $\longrightarrow$ RESPONSE

Fig. 2: Environment-Action coupling for Intelligent Behavior

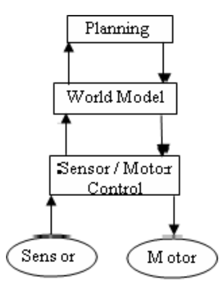

Fig. 3: Classical Control (Deliberative) Model

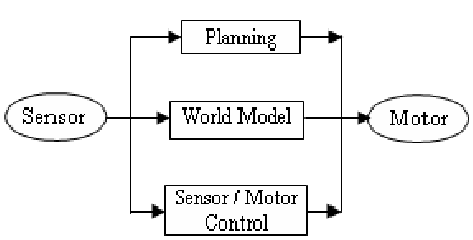

Fig. 4: Behavior Based Control (Reactive)

The Classical Models as told earlier maps the environment first with the help of sensors to some symbolical representation and then takes the decision what to do and sends signals to motors or end effectors. This is known as Deliberative Control. But in Behavior Based Control models the symbolic representation of the action or environment is not much important compared to the previous case. Instead of responding to internal entities, the agent or the robot can respond directly to perception of the real world as shown in Fig. 4. Thus the Reactive Control is best characterized by a direct connection between sensors and effectors/motors.

3. Implementing systems

Two different mechanical systems have been designed and developed for implementing the two different architectures: **ARBIB** (**A**utonomous **R**obot **B**ased on **I**ntelligent **B**ehaviors)**-I**, a simple design and **ARBIB-II**, an advanced version.

3.1. The simple version: ARBIB-I

The mechanical structure of ARBIB-I is very simple and it is completely revealed in Fig. 5. This system uses rechargeable Ni-MH batteries (1.2V, 5000mAH) as power source. A battery bank has been made using these batteries for supplying 12V. Two different Stepper motors (for differential drive) have been used for the movement of the ARBIB-I. As no encoders have been used, stepper motors will help to keep track of the position or speed by its precise rotation in steps. A Philips 8051 microcontroller has been used as the controlling unit of the ARBIB-I. There are three different sensors for the full operation of the robot. IR Sensor measures the distance of the obstacle (range is 30 cm) and passes the information to control unit. There are two IR Sensors fitted at the front, one on the right side and another on the left side of the robot. A battery level detector has been used to indicate the cut-off voltage for the system. The system is programmed to move freely, when no stimuli is present nearby. But if an obstacle comes in front of the robot, or if the battery power is low other behaviors will be suppressed. The microcontroller can be re-programmed for different logic to introduce different behaviors using a serial port. Being the first of its kind in CMERI, Durgapur a wireless miniature camera has been mounted for manual observation.

Fig. 5: The (front) view of ARBIB-I showing IR sensors and a wireless Camera

3.2. ARBIB-II: The advanced design

Though ARBIB-II is the advanced version, but like ARBIB-I, it also has simple mechanical structure with differential driving capability. But the advancement of the design has been incorporated in the sensor suite and sensor fusion. Apart from the IR sensors and battery level detector, here three different sensors have been used for introduction of new behaviors. They are: LDR (Light dependent resistor) for identifying the light sources and helping the system to orient itself properly for recharging the batteries; Whiskers/ Tactiles (touch sensor) to detect obstacles (in addition to IR sensors) which are not detected (out of the sweeping volume due to narrow pencil beam of IR sensors) by the IR sensors; Sound sensor to introduce the behavior of fear. The system uses a 12V Li-Ion battery bank as its power source and another 4V battery bank is used for driving relay and other auxiliary circuitries. Two DC motors differentially drive the system. Fig. 6 shows the front view of ARBIB-II with IR sensors, LDRs and Tactile sensors. The sound sensor is mounted at the back of the robot. ARBIB-II uses a Maxim 8051 microcontroller as its main processing unit due to its large memory capacity. Another 8051 microcontroller is dedicated to send the data of the main processing microcontroller to remote computer wirelessly for post processing.

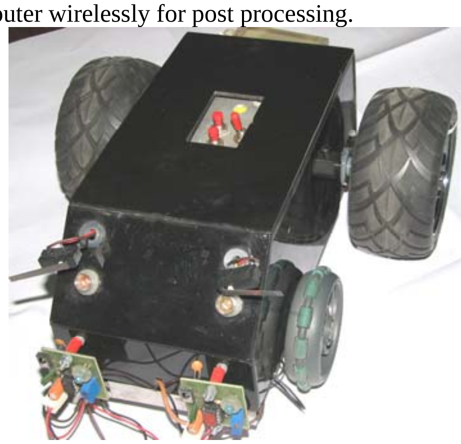

Fig. 6: ARBIB – II: The advanced design

4. The Behavior-based Architectures

In behavior based robotics four architectures are being used popularly over the world. They are Subsumption Architecture ([1], [2], [4], [5]), Action Selection Dynamics [6], Schema-based approach ([7], [8], [9], [10]) Process Description Language [11]. Subsumption architecture is a layered behavior proposed by Brooks. These layers are associated with many simple behaviors. All these simple behaviors combine to form complex behaviors. The layers operate asynchronously. The Action Selection Dynamics proposed by Maes, is an agent based architecture, where it has been proposed to build an intelligent system as a society of interacting, mindless agents, each having their own specific competence. This architecture is an attractive one due to its modularity, distributedness, flexibility and robustness. Agents can be added, changed or modified without caring about the other agents. Ronald C. Arkin used the Schema-based model in a behavior based model. Schemas are coarse-grained functional units, being approximately at the same level of description of ethological and neurobiological units of automatic action control such as detect prey or escape. They do not only contain conceptual knowledge, but are strongly action-oriented and include perceptual and motor elements. Schemas consist of actions and sensory information organized around, and serving to realize, a goal or a set of related goals. In simplest form, schemas can be described as sets of rules having the form condition→ action or condition→action→exception, that can act in parallel or in series, and whose success corresponds to the achievement of a goal. There is only a few work reported in the literature regarding Process description language.

4.1. Subsumption Architecture and the logic of operation for ARBIB-I

The Subsumption architecture as presented by Brooks is a parallel and distributed computation formalism for connecting sensors to actuators in robots mainly the behavior based robots. Subsumption is simply composed of completely separate agent (known as Finite State Machines) with input/output lines that can receive or send messages. To the input line there may be a suppressor (suppresses normal flow of data for a specific time period and allows data from a different agent) and in the output line there can be an inhibitor (inhibits the output of another agent for a specific time period). It is clear from the Fig. 7 that this architecture is consisted of three different layers viz. 'Emergency layer', 'Task layer' and 'Motion layer'. 'Emergency layer' is the high priority layer and any behavior/ activity in this layer will get the first preference for execution. The 'Task layer' is with second priority and the lowest priority layer is 'Motion layer'. As per the Subsumption Architecture is concerned the different behaviors have different independent logic and priority. The behaviors are purely individual entity and they can be plugged to each other to make a complete and robust autonomous system (i.e. can act as Augmented Finite State Machines). The high priority behaviors suppress the low priority behaviors, like the normal behavior of the ARBIB-I is 'Wandering' and it is suppressed by 'Avoid Obstacle' or 'Escape' or 'Follow' or 'Hunger' to avoid obstacles or to run away or to inform the low battery level. The driving logic for the system is as follows. 'Wander' behavior is activated as soon as the system is powered 'ON' and the programme check for any signal from the sensors. If any sensor is active the system responds accordingly depending upon the priority of the task corresponding to that specific sensor. If one of the IR sensors is active the 'Avoid Obstacle' behavior takes care of the robot. The emergency layer consists of the behaviors, 'Struck' and 'Hungry' and these are with highest priority. When the both IR sensors have same analog voltage, the system is supposed to get struck and the controller immediately reverses the motor for 2 seconds. As the ‘Hunger’ has highest priority whatever be the condition of the system, it will override other behaviors/activities to switch on the LED and the beeper and then automatically shuts down the system.

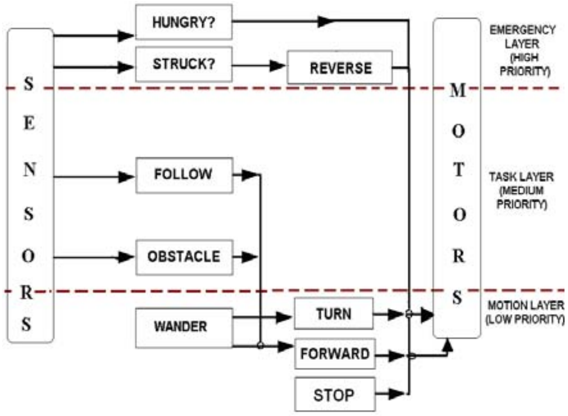

Fig. 7: Subsumption Architecture for ARBIB-I

4.2 Motor Schema Theory and the logic of operation of ARBIB-II

Schema theory is a powerful and expressive means for describing behavior, both neuroscientific and robotic [12, 13]. The concept of schema was initially originated in psychology and neurology. As per the Webster dictionary it means “a mental codification of experience that includes a particular organized way of perceiving cognitively and responding to a complex situation or set of stimuli”. Architectures including distributed and competitive functional units are often referred to as ‘behavior-based’ or ‘schema-based’. Arkin proposed the motor-schema based architecture for autonomous navigation. A successful implementation has been done in modeling a preying mantis.

Fig. 8 illustrates the main functional components of a sample motor schema [14], which is named after its goal: *chase food*. It is triggered by two sample conditions: one if *hungry*, that indicates the value of the drive and *food in sight*, that refers to the presence of specific stimuli in the environment. It also includes three related actions (represented as a localist sub-unity of the *schema): approach food*, *grab food* and *eat food*. These actions can be implemented as rules or set of rules, which receive perceptual input and send motor commands such as “go right’ or ‘go left’. Schemas are preferred due its goal-orientedness, flexibility, selectivity and excitability.

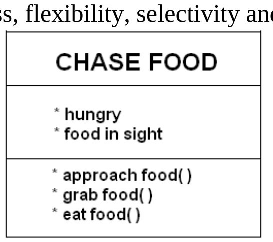

Fig. 8: Example of a sample schema: *chase food*

The motor-schema theory as implemented in ARBIB-II is shown in Fig. 9. Here three major schemas have been used: **Move-to-food**, **Move-away-from-danger** and **Move-to-light**. The robot maintains two internal variables that represent the robot’s hunger and fear. Initially, the values of all these variables are set to zero. The value of the variable hunger increases with time following the relation given in (1):

$$hunger = hunger + (5 * time^3) \quad (1)$$

This conforms to the approximate drainage rate of the battery. **Move-to-food** behavior is triggered by its perceptual schema detect food. The variable hunger reaches the highest value when detect food is activated. The **move-to-food** behavior produces a direction that will move the robot toward the area of largest light. Here food refers to the light (mainly the solar energy) energy for charging the batteries. A solar panel is incorporated on the top of the robot (not shown in the Fig.) for recharging the battery. If the robot reaches an area where the maximum voltage (6 V) is obtained from the solar panel, it is assumed to be in the recharging zone. This is ensured by the generation of same intensity (current) on both the LDRs. If the system remains stopped within the said zone for 100 seconds, the variable hunger is again set to zero.

Fear is another state assigned to the system. The behavior **move-away-from-danger** is activated by its corresponding perceptual schema detect danger. The variable fear reaches its greatest value before detect danger is triggered. This value stays high for 5 seconds and then again set to zero. **Move-to-light** is an anti-moth behavior of the robot due to the perceptual schema detect light. No variable is associated with this schema; rather it is closely associated with **move-to-food** behavior. The values of the variables are used by the action selection module to select the appropriate action to be chosen by the robot’s processor. The motivational variable with highest current value is always chosen. If there is an associated stimulus present, such as food for the hunger variable, then the output of the corresponding behavior is sent for action. If there is no associated stimulus visible, then the process is repeated with next higher value. If there are no stimuli present, the action selection mechanism does not send any command to the robot. There is no predetermined hierarchy or layering; the action chosen depends directly upon the value of the motivational variables and visible stimuli at that moment in time.

For example, if the current values of the motivational variables, hunger and fear be 200 and 80, then if food is visible in the environment the **move-to-food** behavior will be activated. Now if the values of the variables are same as above, but there is no food nearby, then **move-away-from-danger** behavior will be triggered.

The remaining part of the model as shown in the right side in Fig. 9 is a colony style-architecture with three priority levels. The output of the higher level suppresses the output of lower level for a specific amount of time (5 seconds). The lowest level is **move-forward** (wonder) behavior. This behavior moves the robot in forward direction constantly in search of stimuli. If there comes a stimulus, hunger or fear the action-selection produces a different command (left or right turn or stop) and suppresses the **move-forward** behavior. There is a continuous checking in a loop for the stimuli and **move-forward** behavior is suppressed till there is a stimulus. As soon as a stimulus is vanished, the **move-forward** behavior is again activated. The **obstacle avoidance** behavior has the highest priority. When this behavior is active, all commands from the other low level behaviors are suppressed. The **obstacle avoidance** behavior causes the robot to either take a left turn (if the obstacle found on the right side) or a right turn (obstacle on right side) or makes it to STOP, then move backwards and take a left turn (if obstacle is in front i.e. both the left and right IR sensors are obstructed).

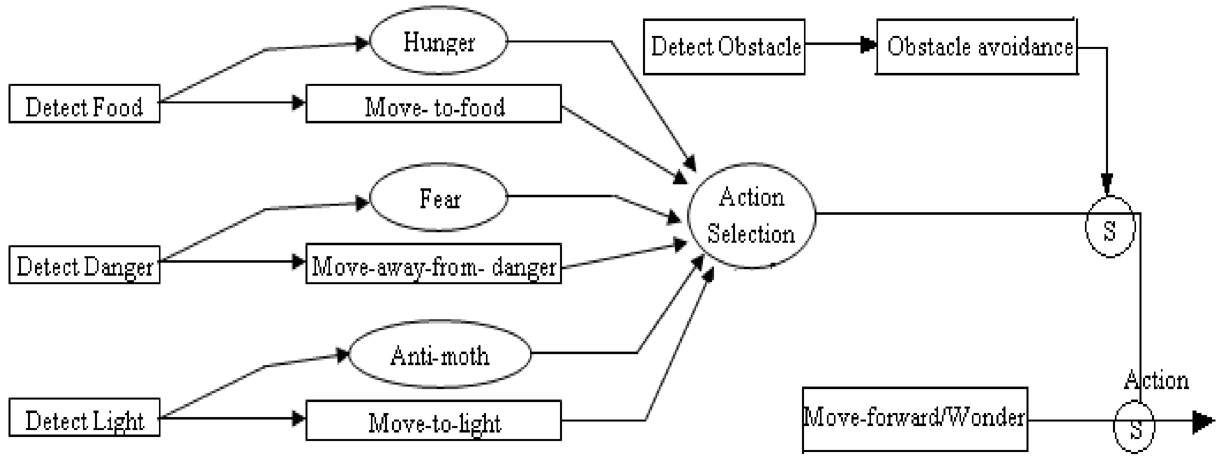

Fig. 9: The motor schema theory implemented on ARBIB-II with action-selection arbitration

The motor schema theory along with the colony-style architecture has been implemented in ARBIB-II incrementally. First the **move-forward** behavior was created. Then the **obstacle avoidance** behavior has been added.

1. Increment hunger and set fear.

Hunger = hunger + $(5 \times \text{time}^3)$ /* increment hunger is non-linearly increasing with time*/
If danger is detected,
then fear = 5000 ; /* set fear at highest level*/
else fear = 0; /* reset fear when no danger is sensed*/

2. Check if food is close enough to eat.

If food is obtained,
then hunger = 0; /* reset hunger (reached the area where the voltage from solar panel is maximum)*/

3. Each behavior produces a direction or Stop command, based on the input from its corresponding perceptual schema.

(a) move-to-food, move-to-light

If food/light is on right side,
Then output forward right;
If food/light is on left side,
Then output forward left;
If light is same on both the sides,
Then output forward;

(b) move-away-from-danger

If danger is detected,
Then output move fast;
Else NO-FEAR;

4. Choose an output from a behavior, to pass along to the robot

If there is an associated stimulus for the motivational variable with greatest value,
Then output direction from behavior corresponding to this variable,
Else if there is an associated stimulus for the motivational variable with second greatest value,
Then output direction from behavior corresponding to this variable,
Else do nothing;

Fig. 10: Algorithm associated with the schema-style architecture running on ARBIB - II

Next the food searching behavior was incorporated. When all these behaviors were working fine, the fear and anti-moth behavior had been introduced and thus developed an advanced and little complex system.

5. Experimental result

To test both the systems for autonomous behavior based exploration, same pattern of obstacles were used. In case of ARBIB-I the exploration is very simple. As shown in Fig. 11 (from left to right) it moves smoothly avoiding the obstacles on left and right. The sequences 2 -3 and 5-6 of Fig. 11 show the right turn and left turn respectively, which is the output of the behavior **obstacle-avoidance**.

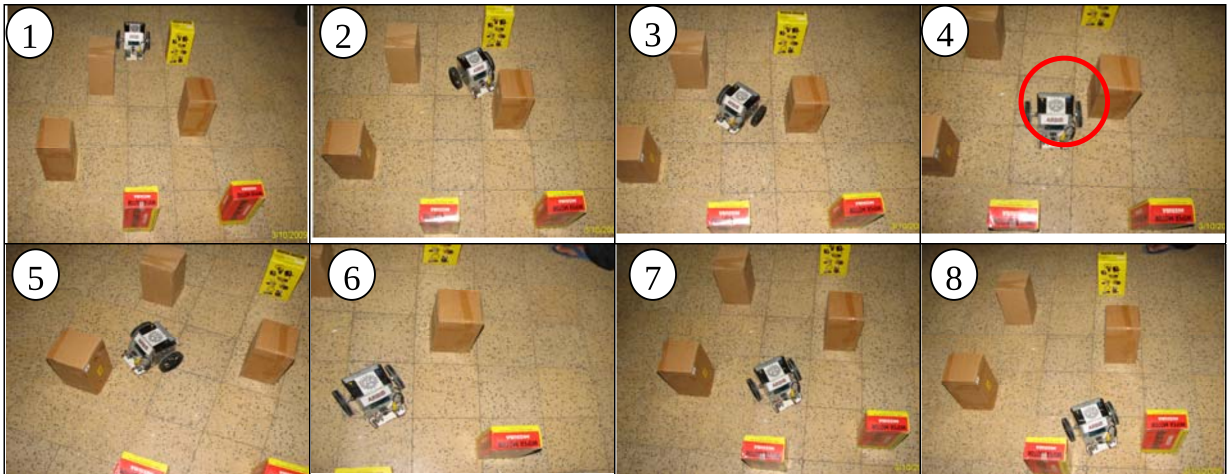

Fig. 11: The sequences of pictures show the operation during one experiment. It proceeds from left to right. At first, ARBIB-I starts exploring the terrain. Due to the obstacle on the right, ARBIB-I takes a left turn. Again the obstacle on the right side makes the system to take turn to the left. The system comes out at last safely.

The same pattern of obstacles has been used for ARBIB-II also. Initially all the variables used in this system is set to zero. Hunger increases with time and if there is a danger only the fear increases. Hunger again sets to zero if the robot stays in stopped condition for 100 seconds in a recharging zone as shown in Fig. 12 with red circle. The danger is responded by ARBIB-II using lighting a red LED mounted at the back (as shown in Fig. 13) and simultaneously increasing the speed of the system than normal time.

The following is a description of one particular execution sequence, shown in Fig. 14. At first ARBIB-II starts from the starting position. When an obstacle come on the right (in second picture), it takes a left turn and gradually comes out. This clearly shows the **obstacle avoidance** behavior of ARBIB-II.

The Anti-moth behavior has been clearly depicted in Fig. 15 (a – c). Initially there is no light source (stimulus) present near the system. When a light source comes on the left side (in the second picture), the robot takes a left turn to maximize the light intensity. Again if a light source is present on the right side, ARBIB-II takes a right turn.

The **move-forward**/ **wonder** behavior is active in both the cases from the starting and shown in the first pictures of Fig. 11 and 14.

6. Comparison

Both the architectures have been implemented successfully on two different models, ARBIB-I and ARBIB-II.

Fig. 12: The variable hunger sets to zero when the solar panel generates maximum voltage in an enlightened zone (in red circle)

Fig. 13: Blinking of red LED at the back depicts that danger is present and ARBIB-II shows the ‘fear’ behavior by speeding up than normal

It is clear from the above discussion that Subsumption architecture is a single stage, parallel layer architecture.

Fig. 14: These sequences of pictures show the operation of ARBIB-II during the autonomous behavior-based exploration using the same pattern of obstacles as used in case of the previous experiment with ARBIB-I. The sequences proceed from left to right. ARBIB-II also responds to left and right turning due to the presence of obstacles on the left and right respectively. The robot navigates safely at last.

Fig. 15(a): Wondering behavior of ARBIB-II without any stimulus. It follows a straight path

Fig. 15(b): Anti-moth behavior of the system helps it to take a left turn due to presence of a light source on the left

Fig. 15(c): System takes a right turn due to presence of the stimulus - light on the right side of the system

It is a very simple architecture for implementation in behavior based systems. But complex or advanced systems (learning systems) are difficult to develop using this architecture. Mainly three layers are present in this architecture; Emergency layer (highest priority), task layer (medium priority) and motion layer (low priority). The behaviors have been categorized under the above priority classification. Here no action-selection arbitration is present. The available stimulus present within the environment triggers the relevant behavior. Here the priorities of various layers/ behaviors are assigned already with the algorithm.

The Motor schema theory is a concept originally generated in the psychology and the neurology. In Motor schema theory used for behavior-based systems, there are two different stages and both are with parallel layers. Stage-I is related with the detection of behaviors and selection of the behaviors depending upon the current accumulated values. This is more dynamic and rational than Subsumption architecture. Priority for this stage-I is not predefined in the algorithm as in the case of Subsumption architecture. Stage-II is the prioritization stage of the behaviors. The highest priority is assigned to obstacle-avoidance behavior (like the emergency layer in Subsumption architecture). Next priority is given to the action selected by the action-selection arbitration. The action/ motion layer has got the lowest priority. This stage-II is similar to the Subsumption architecture. This theory for its double layered configuration can be used for the development of various advanced behavior based system especially with learning.

7. Conclusion and Future work

The Subsumption architecture is a parallel and distributed computation formalism for connecting sensors to actuators in robots mainly the behavior based robots. Subsumption is simply composed of completely separate agents. This architecture is a very simple, single stage, parallel layered architecture to implement in robotic systems. The Motor schema theory is a double stage parallel layered architecture for development of the advanced robotic systems, especially for the learning robots. The first stage may be used for gathering the related information for learning and the stage-II for implementation of the learning algorithm. It is irrelevant to draw a specific conclusion after the comparison of these two architectures for autonomous behavior-based exploration. Both of them worked fine and produced a good result for autonomous behavior-based exploration. But they differ when they are used for implementation in learning systems. However, Motor schema theory, though may look complex, is good for handling large numbers of complex behaviors. Work is in progress for implementing learning to behavior-based systems using both these two architectures.

Acknowledgment

The authors are grateful to two undergraduate students, Ms. A. Mandal and Ms. B. Das for their tireless work to successfully complete the project. Authors also thank the Director, CMERI and other CMERI personals who have directly or indirectly helped to complete this work successfully.

References

[1] Rodney A. Brooks, *Cambrian Intelligence the Early History of the New AI*, The MIT Press, 1999

[2] R. Brooks, "Intelligence without reason", *Proc. of the Conference of International Joint Conference on Artificial Intelligence (IJCAI)*, 1991

[3] D. N. Ray, "Behavior Based Robotics: Steps towards Intelligence", *The Bulletin of Engineering and Science*, Vol. 1, No. 2, Page 54-60, 2006

[4] Rodney A. Brooks and Stein L. Andrea, "Building Brains for Bodies", A.I. Memo No. 1439, 1993

[5] R. Brooks, "A Robust Layered Control System for a Mobile Robot", *IEEE Journal of Robotics & Automation*, Vol. 2, No.1, Page 14-23, 1986

[6] P. Maes, "The Dynamics of Action Selection", *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 991-997, 1989

[7] M. Arbib, *Schema Theory*, In Encyclopedia of Artificial Intelligence, 2nd Edition, Vol. 2, pp 1427-1443, Jhon Wiley and Sons., New York, 1992

[8] Ronald C. Arkin, "Motor Schema — Based Mobile Robot Navigation", *The International Journal of Robotics Research*, Vol. 8, No. 4, 92-112, 1989

[9] Ronald C. Arkin, *Behavior-Based Robotics*, The MIT Press, 1998

[10] R.C. Arkin, Integrating Behavioral, Perceptual, and World Knowledge in Reactive Navigation, *Journal of Robotics and Autonomous Systems*, Page 105-122, 1990

[11] L. Steels, A Case study in the behavior-oriented design of autonomous agents, *Proc. of 3rd International Conference on Simulation of Adaptive Behavior*, The MIT Press/ Bradford Books, 1994

[12] Khaled S. Ali and Ronald C. Arkin, "Implementing Schema-theoretic Models of Animal Behavior in Robotic Systems", *Proc. of the international workshop on advanced motion control*, pp. 246-253, 1998

[13] R. C. Arkin, "Motor schema based navigation for a mobile robot: An approach to programming by behavior", *Proc. of IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 264-271, 1987

[14] G. Pezzulo, "Schemas and Schema-based Architectures", http://www.istc.cnr.it/doc/1a_20070514155535t_schema.pdf