The Mobot Architecture: A Design of Autonomy Data Evaluation

Arshad Mansoor, Adeel Ahmed Hashmi, and Nawazish Khokhar

***Abstract*—Most large, long term mobile robot systems start with some architecture. We studied some of the architectures which are implemented on robot previously, the advantages and disadvantages and finally select siemens 4 view architecture, the best possible architecture for mobile robot. In the designing phase of developing architecture of a robot using siemens 4 view, first of all we will identify the quality goals in the form of factor tables. Quality goals/Factor tables are basically broader statements which gave the information of the attributes, which this system will be capable to perform at the end and it will also provide different qualities that the final system should exhibit and are used to guide the development process. We will only identify some of the quality goals like navigation, security, monitoring, communication etc. Complete and detailed architecture will be developed in future. We named this architecture as Mobot.**

***Index Terms*—Quality goals, factor tables, issue card, siemens 4 view.**

I. Introduction

Over the years the field of mobile robot yielded many architectural proposals. Autonomous mobile robots are being developed with the aim of accomplishing complex tasks in different environments, including human habitats as well as less friendly places, such as distant planets and underwater regions. A major challenge faced by such robots is to make sure that their actions are executed correctly and reliably, despite the dynamics and the uncertainty inherent in their working space [1]-[5].

In this paper we generally develop architecture for an autonomous mobile robot and it fulfils all the requirements which are given to us.

This problem focuses on how to mapping of robot architecture on semen's four view. The mobile architecture must deal with the sensors attached to it and they must react in time with the activities of the system in its environment. The following activities which mobile robot normally has to carry out: [3]-[10]

- Get input from the sensors.
- Controlling the runtime actions of its wheels and other moveable parts,
- Planning the path which it follows.

A number of factors relating to the tasks: [10]

- Obstacles can come in the robot's path.
- Behaviour
- Optimized path finding
- Navigation
- The input of sensors may have problem.
- Low power problem.
- Mechanical limitations may restrict the accuracy with which the robot moves.
- The robot may manipulate hazardous materials.
- Unpredictable events may leave little time for responding.

II. Design Considerations

We concluded the following basic requirements for the Mobot's architecture.

The architecture must accommodate deliberative and reactive behaviour. The robot must do the actions and these actions must be coordinated in such a way that they can achieve the desired objective as directed (e.g., collect a book from library or any objective which is given to it or sample of soil from other plant) with judging the hurdles comes in its way by the external surroundings (e.g., if any obstacle then turn or jump) [10]-[13].

The architecture of the robot must cater the unexpected situations which come in its way to achieve the objective. As it is impossible to provide the data of all the circumstances/situations as they are not predictable before their occurrence (e.g. suddenly some one comes in its way or any hurdle like big stone comes in its way). The architecture of robot must provide a framework that support partial or undependable information (e.g., information from sensors) [12]-[15].

The architecture of robot also has the capability to handle the different problems that are critical to the robot's operation and its external surroundings. When a robot has this capability of defect tolerance, security, and different performance attributes then this architecture will help in maintaining the reliability of the robot, its administration, and its external environment (e.g. Problems like low power battery, uncertain situations, or suddenly opening of doors will not create any unexpected situation for the robot) [13], [16].

The architecture must give the flexibility in design so that any new technology or module can be incorporated effectively and efficiently. Moreover, with the change in the goal of the robot, the configuration and modification in the architecture will also be changed. Robot must have the capability to take decision at runtime and adopt changes as and when required [13], [17], [18].

Fig. 1 shows the abstract level of the architecture and these are some of the requirements which are totally depend upon the tasks which are to be performed by this robot. If task is complex obviously the degree of complexity will also increase. External environment in which this robot will perform the tasks will also play a vital role as if

Manuscript received November 19, 2012; revised February 20, 2013.
The authors are with Shaheed Zulfikar Ali Bhutto Institute of Science and Technologies, Islamabad, Pakistan (e-mail: flt_lt_arshad@hotmail.com, a_a_h82@hotmail.com, nawazish_k@hotmail.com).

DOI: 10.7763/IJCEE.2013.V5.765

environment is predictable then architecture will be simple and vice versa. For instance, if robot has to performed task on other plant for sampling of the soil then we must know that if any fault arises then it is difficult to get back to the maintenance facility so we have to make it capable enough to trouble shoot the faults by itself [13], [19].

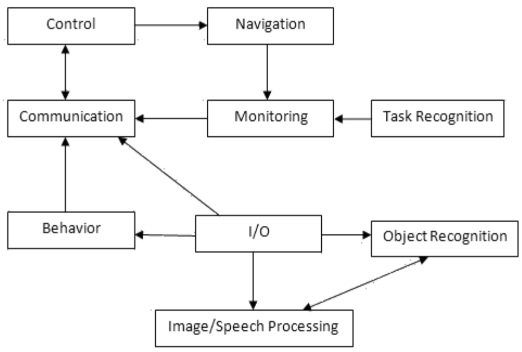

Fig. 1. Abstract view of architecture

III. Global Analysis

In this paper we only identified some of the quality goals which are very essential for the robot to handle [13], [20]-[22].

A. Navigation:

TABLE I: Factor Table: Navigation

Factor	Flexibility	Impact
P1.Navigation		
P1.1 Velocity Calculation and Control		
Robot must be able to calculate the velocity according to the situation	Not Negotiable	High
P1.2. Turning Angle Calculation and Control		
Robot must be able to calculate the angle according to the situation	Not Negotiable	High
P1.3. Wheel Movement		
Robot must be able to calculate the movement of the wheel according to the situation	Not Negotiable	High
P1.4. Sensor Command and Control		
Sensors like camera must move according to the movement of the robot.	Not Negotiable	High
T1. Navigation		
T1.1. Hardware of Motors		
Hardware of motors which will drive the wheel, sensors and other components.	Different types of hardware are there to use.	High

B. Security:

TABLE II: Factor Table: Security

Factor	Flexibility	Impact
P2. Security		
P2.1 Encryption and Decryption		
System support these techniques for the signals	Negotiable	High
T2. Security		
T2.1. Hardware accelerated security		
Hardware accelerated encryption and decryption are more secure	Negotiable	High
O2. Security		
O2.1. Which technique for encryption and decryption		
If there is any organizational limit for which technique is to be used.		

C. Monitoring:

TABLE III: Factor Table: Monitoring

Factor	Flexibility	Impact
P3. Monitoring		
P3.1 Health Monitoring		
Robot must check itself about its health	Not Negotiable	High
O3. Monitoring		
O3.1. Which parameters to monitor		
If there is any organizational limit to monitor some specific parameters.	Negotiable	Low

D. Communication

TABLE IV: Factor Table: Communication

Factor	Flexibility	Impact
P4. Communication		
P4.1 Protocol selection		
Robot must be able to communicate using some standard protocol	Negotiable	High
P4.2. Communication Mode for Information		
Transmutation of information	Not Negotiable	Medium
P4.3. Communication Mode for commands		
Tasmanian of commands and there response must be done in an encrypted way and type of communication must be point to point	Non negotiable	High
T4. Communication		
T4.1. Hardware for specific communication		
Hardware to receive and transmit information	Non Negotiable	High
T4 Communication		
T4.2 Software for communication		
Which implementation of protocol stack to be used	Negotiable	High
O4 Communication		
O4.1 preferred Protocol		
Preference for communication style	Negotiable	High

E. Intelligence:

TABLE V: Factor Table: Intelligence

Factor	Flexibility	Impact
P5. Intelligence		
P5.1 Input Recognition		
P5.1.1 Tasks and Command Recognition		
Robot must differentiate between the command and task	Not Negotiable	High
P5.1.2 Control and Command signals Recognition		
Robot must differentiate between the command and control signals	Non negotiable	High
P5.2 Map Construction		
P5.2.1 Reconstruction of Area Map		
Robot must process and reconstruct the map	Negotiable	High
P5.3 Navigation Path Calculation		
Calculate the optimized path for completion of task	Negotiable	High

F. Behaviour:

TABLE VI: Factor Table: Behaviour

Factor	Flexibility	Impact
P6. Behaviour		
P6.1 Decision making logic		
Robot must be prompt to take decision	Negotiabl e	High
P6.2 Task Execution		
Execute the task as directed to him	Non negotiable	low

G. *Input / Output:*

TABLE VII: FACTOR TABLE: INPUT/OUTPUT

Factor	Flexibility	Impact
P7. Input/Output		
P7.1 Processing of different sensors input		
Receive data from sensors and then process the inputs	Semi Negotiable	High
T7. Input/Output		
T7.1 Hardware for sensors		
Hardware for the sensors	Non Negotiable	Low
O7. Input/Output		
O7.1 Data Availability		
Data from the sensors will be available every time	Negotiable	High

IV. ISSUE CARD

In this section we will describes the issue tables that will help us to identify the factors which will control the design and the strategies of the architecture. We will only cover some of the issues in this paper and detailed issues will be discussed in future. A unique identifier will be used for each issue table as a label. We at this time only identify some of the issues relating to the factors which have been identified before [13], [23].

TABLE VIII: ISSUE CARD

Issue: Response time is critical
As most of activates are dependent upon the processing and calculations so the time of getting result is critical and must be fast. Influencing Factors P1.1 Velocity Calculation and Control P1.2. Turning Angle Calculation and Control P2.1 Encryption and Decryption P3.1 Health Monitoring P5.2 Map Construction P5.3 Navigation Path Calculation P7.1 Processing of different sensors input
Solutions The robot must have high processor and memory so that it can easily process the request and processing time must be very fast hardware encryption and decryption is recommended. Strategy Use multiprocessor technology Strategy Size of RAM must be Large Strategy Wireless network is recommended

V. SIEMENS 4 VIEW

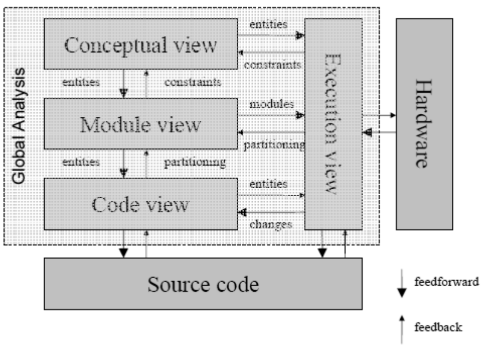

Fig. 2. Siemens 4 View

Siemens 4 view is one of the best practices which are to be used for any architecture as it is clear and in detail. It consist of four views [2], [4], [5], [12], [13], [16]

- Conceptual View
- Module View
- Execution View
- Code view

In this paper we only show the high level or abstract level of these views. When we implement the siemens 4 view the new figure will be Fig. 3.

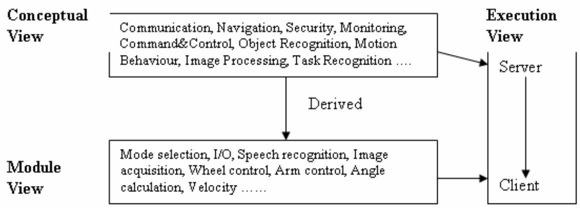

Fig. 3. S4V Abstract Level

In Fig. 3 we implement conceptual, module and execution view, as time was the constraint so we only implement the abstract level of these three views.

VI. CONCEPTUAL VIEW

A number of methods or models are given to use in this view: construction decomposition, functional decomposition, class or object decomposition, other decompositions (power, resources, recycling, maintenance, project management, cost, ...), and related models (performance, behaviour, cost, ...); allocation, dependency structure; identify the infrastructure (factoring out shareable implementations), classify the technology in core, key and base technology; integrating concepts (start up, shutdown, safety, exception handling, persistency ,resource management, ...)[13].

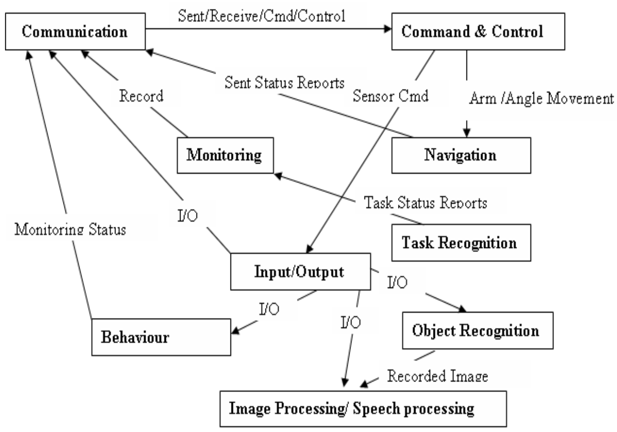

Fig. 4. Conceptual View

VII. MODEL VIEW

The module view of the robot architecture shows how the software components will be mapped into different modules, sub modules and subsystems. It will also tell us about the runtime components used in the architecture. In this paper we only discuss two modules i-e communication and input/output.

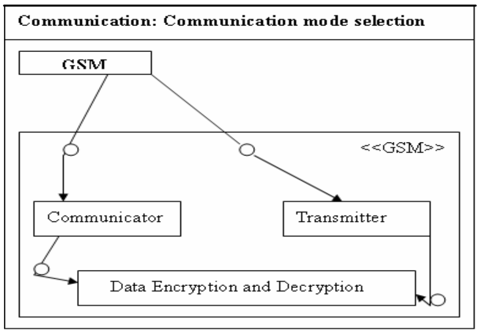

Fig. 5. Subsystem of Communication

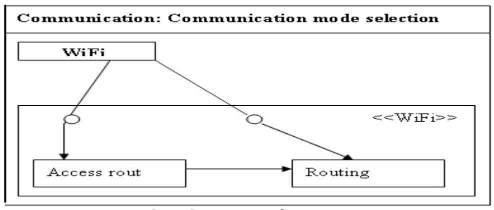

Fig. 6. Sub Subsystem of Communication

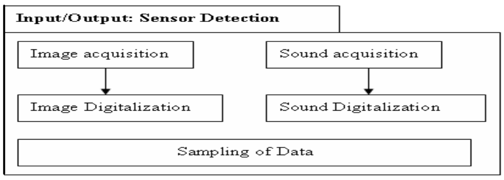

Fig. 7. Subsystem of Input / Output

VIII. Execution View

The execution views of software architecture talk about the hardware mapping with the modules, sub modules and subsystems. It also tells how much physical memory is required, how much speed of processor will be required and how the communications will be done. This view also gives the information of the entities used during run-time, their respective attributes and their mapping with the hardware. This view also gave us the details of the flow of controls from hardware to the software and how the instructions will be executed [24]-[27].

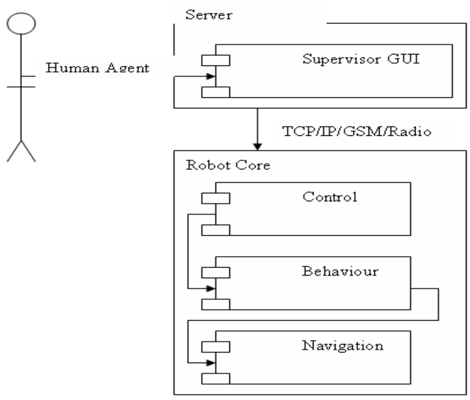

Fig. 8. Execution view

IX. Conclusion

In this paper we have described siemens 4 views and used it for the design of architecture of Mobot. An open issue addressed by this paper is how we can effectively design mobile architecture using siemens 4 views. After this paper we are very clear that to used this approach we need ample time as it is detailed.

References

[1] Bushmann, *Pattern Oriented Software Architecture: A System of Patterns*, Wiley, 1996

[2] Champeny-Bares, Coppersmith, and Dowling. (1993). The terregator mobile robot. *Technical Report CMU-RI-TR-93-03*. Robotics Institute, Carnegie Mellon University, Pittsburgh, PA. [Online]. Available: http://asimo.honda.com http://www.sony.net/Products/aibo

[3] Elfes, "Sonar-based real-world mapping and navigation," *IEEE Journal of Robotics and Automation*, no. 3, 1987, pp. 249-265.

[4] G. Podnar, K. Dowling, and M. Blackwell, "A functional vehicle for autonomous mobile robot research," Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1984.

[5] C. Hofmeister, R. Nord, and D. Soni, "Applied software architecture," Addison-Wesley, Reading, MA, 2000.

[6] J. Castro, and J. Kramer. Proc. of the First Int. Workshop from Software Requirements to Architectures (STRAW'01, Toronto, Canada. (14 May 2001). IEEE Computer Society Press. [Online]. Available: http://www.cin.ufpe.br/ , straw01/

[7] R. Allen and D. Garlan, "A formal approach to software architectures," in *Proc. 1992 IFIP Congress*, September 1992.

[8] R. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal of Robotics and Automation*, vol. 2, no. 1 pp. 14-23, 1986.

[9] R. Alami, R. Chatila, S. Fleury, M. Ghallab, and F. Ingrand, "An architecture for autonomy," *International Journal of Robotics Research*, vol. 17, no. 4, pp. 315-337, 1998.

[10] M. Shaw, D. Garlan, R. Allen, D. Klein, J. Ockerbloom, C. Scott, and M. Schumacher, "Candidate model problems in software architecture," The Software Architecture Group, Computer Science Department, Carnegie Mellon University, Pittsburgh PA 15213 , Version 1.3: January 1995.

[11] E. Gat, "On three-layer architectures," *A.I. and Mobile Robots*, D. Korten Kamp *et al*. Eds. MIT/AAAI Press, RR. N°3552, 1997.

[12] Georgia Tech Mobile Robot Laboratory. (2002). MissionLab: User manual for MissionLab version 5.0. [Online]. Available: http://www.cc.gatech.edu/ai/robotlab/ research/MissionLab/, Georgia Institute of Technology.

[13] J. J. Borrely *et al*., "The Orccad Architecture," *International Journal of Robotics Research*, Special issues on Integrated Architectures for Robot Control and Porgramming, vol. 17, no. 4, pp. 338-359, 1998.

[14] S. Anderson. (2007). Asimov's 'Three Laws of Robotics' and Machine Metaethics. AI and Society, Springer. [Online]. Available: http://en.wikipedia.org/wiki/Three_Laws_of_Robotics for a summary discussion of all 4 laws.

[15] R. C. Arkin, *Behavior-Based Robotics*, MIT Press, Cambridge, Massachusetts, USA, 1998.

[16] R. C. Arkin, "Motor schema-based mobile robot navigation," *The International Journal of Robotics Research*, vol. 8, pp. 92–112, 1989.

[17] T. Braunl, *Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems*, Springer-Verlag, New York, 2003.

[18] R. Bonasso, D. Kortenkamp, D. Miller, and M. Slack, "Experiences with an architecture for intelligent, reactive agents," *Journal of Artificial Intelligence Research*, vol. 9, no. 1, 1997.

[19] J. De Schutter, J. E. Rutgeerts, F. Aertbelien, De Groote, T. De Laet, T. Lefebvre, W. Verdonck, and H. Bruyninckx, "Unified constraint-based task specification for complex sensor-based robot systems," in *Proc. Int. Conf. Robotics and Automation*, Barcelona, Spain, pp. 3618–3623, 2005.

[20] B. Koninckx, "Modular and distributed motion planning, interpolation and execution," Ph. D. thesis, Dept. Mechanical Engineering, K.U.Leuven, Leuven, Belgium, 2003.

[21] OMG. Uml specification. [Online]. Available: http://www.omg.org/technology/documents/formal/uml.htm.

[22] G. E. Reeves. What really happened on mars rover pathfinder. *The Risks Digest*. 19. [Online]. Available: http://catless.ncl.ac.uk/Risks/19.54.html# subj6, 1997

[23] T. Gustav, "Fuzzy similarity-based image processing," Doctoral Dissertation, 2005.

[24] B. Abdelbaki, "Robust execution of robot task-plans: A knowledge-based approach," Doctoral Dissertation, 2008.

[25] Futrell *et al*., *Quality Software Project Management*, 4th ed. Upper Saddle River, NJ: Prentice Hall PTR, 2002.

[26] J. Clifford, A. Dobre, J. Firanski, C. Kirby, M. Towhidnejad, and C. Reinholtz, *Blackbird UAS Project Description*, May 28, 2008, Proof.

[27] M. Kim, S. Kim, S. Park, M. T. Choi, M. Kim, and H. Gomaa, "UML-based service robot software development: A case study," in *Proc. 28th International Conference on Software Engineering*, Shanghai, China, 2006.

Arshad Mansoor was born in 1978 in bhakkar, Pakistan and received his MS degree from Shaheed Zulfiqar Ali Bhutto Institute of Science and Technology (SZABIST) University.

He is working as senior System Analyst in PAC. He has keen interest in Software Engineering area and has publications in this area.

Adeel Ahmed Hashmi was born in year 1982 in Lahore, Pakistan. The author did his graduation in 2004 from NUST, Pakistan in Information Technology and his MS in year 2010 in Software Engineering from SZABIST, Pakistan.

He has been working as a research associate in listed universities and working as a consultant with software companies in Pakistan. Previous publication includes research on multi agent systems e.g . MAGISTIC: An Automated Approach to Future Business , 4th International Conference on New Trends in Information Science and Service Science (NISS2010) Gyeongju, Korea, Electroencephalographic e.g Classifying Of Electroencephalographic Data a Rhythmic Model – 1.Conference in Venice, Italy November 24-26, 2010 (http://www.waset.org/journals/waset/)

Nawazish Khokhar was born in 1979 in Pakistan and did his MS degree from Shaheed Zulfiqar Ali Bhutto Institute of Science and Technology (SZABIST) University.

He has done courses in HR Management and is currently working on research projects related to quality assurance and control. He has keen interest in Software Engineering area and has publications in this area.