Open Modular Design for Robotic Systems

Ioannis Chochlidakis, Yiannis Gatsoulis and Gurvinder S. Virk

Intelligent Systems Group, Department of Mechanical Engineering, University of Leeds, Woodhouse Lane, Leeds, West Yorkshire, UK
men2ic@leeds.ac.uk
menig@leeds.ac.uk
g.s.virk@leeds.ac.uk

Abstract.

Reconfigurable robots offer a new approach in robotic systems so that particular solutions can be easily put together by integrating suitable modular components. The CLAWAR approach to modular design using concepts of basic modules and super-modules to the cooperative systems are extended to develop a generic methodology for the design environment. A case study is also presented for designing climbing robots for biomedical applications based on this methodology.

1. Introduction

A modular reconfigurable robotic system is a collection of individual components that can form different robot configurations and geometries [3] for specific task requirements. The CLAWAR community has proposed the development of basic modules [18] (Input, Output, Processing, and Infrastructure) and super-modules that can integrate to each other via the interaction space (comprising power, databus, environment, digital, analogue and mechanics). These concepts are used here to propose the development of software design tools to assist robotisists to realise effective solutions by selecting components from a library of modules. In this way additional modules may be created to enhance the database.

Compared to a conventional robot system whose design is based on the use of specific parts and processes peculiar to the designer to an open modular designed system offers many advantages; these include the following:

- All the modules do not need to be designed from scratch but chosen from those that already exist. Such reuse will allow much faster and cheaper designs to be realised.

- The design can focus on the new modules needed allowing more time if necessary to realise better performances.
- The design is flexible and can be easily adapted to changing needs.

Of course for this work, a viable framework that is acceptable to the wide community needs to exist. For this reason the CLAWAR community has been working on developing robot component modularity and our desire to support this work.

2. Requirements of Modular Robots

Compatibility among the modules is a critical issue especially when tasks such as fire-fighting, urban search and rescue after earthquakes or natural disasters and battlefield reconnaissance where robots face unexpected situations and many kinds of obstacles [11] while in need to fulfil their complicated mission. The compatibility among connectable modules is a result of the design and implementation of the connectors. These connectors allow a module to connect and disconnect with one another in a simple way. Hence, the mechanics of the connector modules must accomplish the following four requirements.

a. Power efficiency – as limited on-board power supplies exist at present

b. Reliable – connectors must endure various operations

c. Compact (Miniaturization) – the mechanism must fit into a tight place

d. Designed with open modularity – newly designed modules should be able to connect to older designs

The need for structured modular approach for robot design is a major factor among cooperating supply-chain based companies as it speeds up the whole NPD (New Product Development) process. For medical robots, even for the external non-miniature ones used worldwide a certain number of requirements are common even though most of the time requirements depend on the environment of use [1, 8 and 12]. The four main modules of any modular robot design have been classified by the CLAWAR community as *input* modules that have to do with inputting information to the robot, *output* modules that provide the interaction to the robot's environment and users, *processing* modules which represent the decision making processes with the robot (normally the software algorithms) and finally *infrastructure* components which support the overall processes needed by the robot (e.g. power, materials, communications, etc).

All robotic systems have specific requirements but when considering a generic design environment (Fig. 1) for a specific application there are two levels that the applications is divided; System Level [17] and Module Level [18]. We need to look at how the basic modules (input, output, processing and infrastructure) can be provided by the available software tools. To cover a great range of modules various robotic applications were studied (Wheeled [13], Climbing [2, 6, 10, 14, and 15], Crawling [3, 14] and Walking [7, 15])

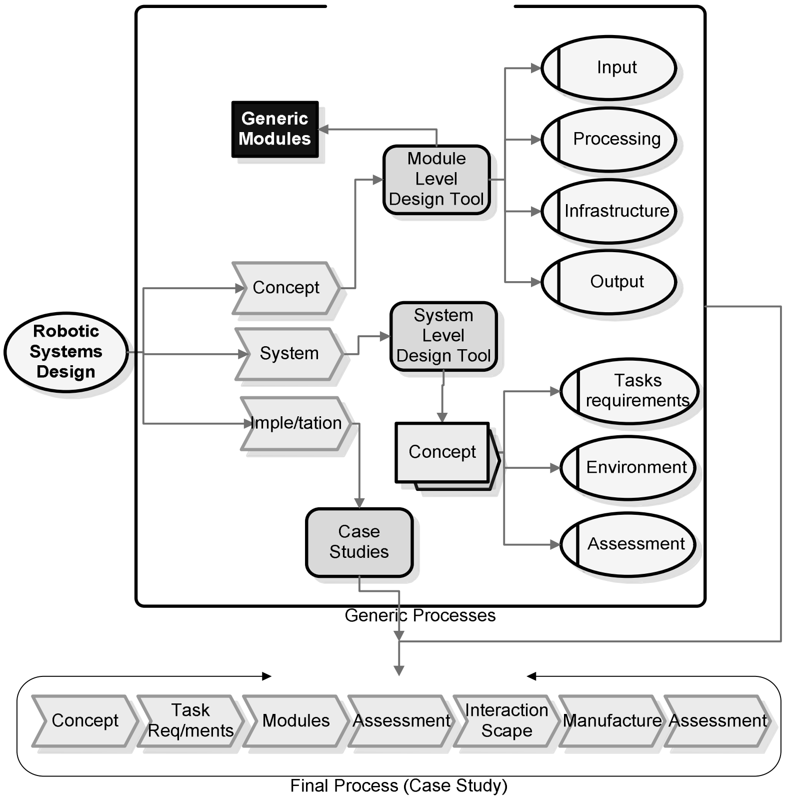

Fig. 1. Overall Generic System Design

Input modules: Many input modules are needed to allow data and information and commends to be inputted into the robot. A few examples are sensors, user commands [4] and GPS. Sensors are the modules that provide systems with the required data; they can be of many types and have a number of attributes (data type, range of operation, environment [4, 11], speed of response, etc). A navigation module can give the ability to navigate the system manually while a GPS can perform this automatically by providing the localisation.

Processing modules: Processing modules are normally non-physical modules which implement and process the information received to perform decision making. Examples of such modules include data analysis, information manipulation, signal processing, learning and navigation.

Infrastructure modules: Infrastructure modules provide some support function. This can be on the mechanical support side, powering aspects or maintenance of the system. Examples of infrastructure modules include materials, power, computing hardware, etc.

Output modules: Output modules provide the direct response to the environment and the user. These modules include actuators/ manipulators for handling or delivery of objects, user feedback, and collaboration with other robots, etc.

The generic basic modules in a generic robotic system are shown in Fig. 2 in a form of a tree. These can be group to group a range of super-modules [7-10].

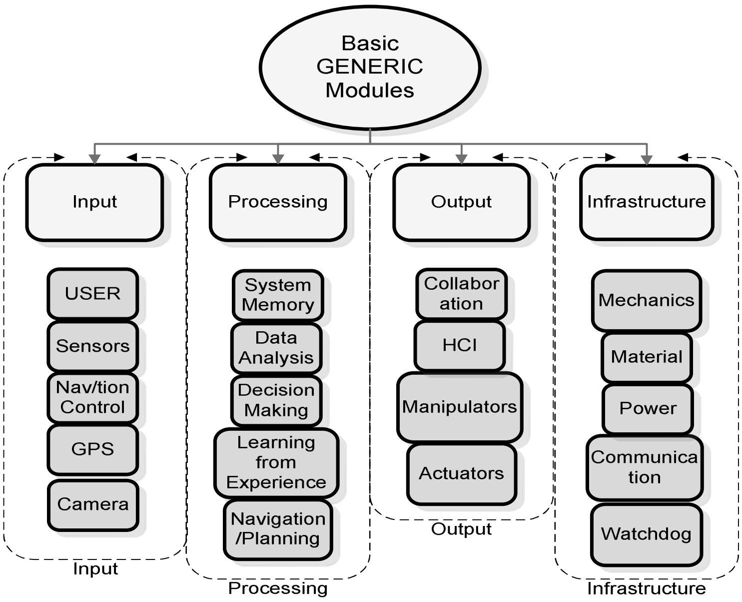

Fig. 2. Generic Modules

3. Software Selection

It is vital to be able to design and simulate the operational environment that the system will work in; hence the need for good design and simulation tools. This should state the selection of possibly several software packages with the required "bridges" or "plug-ins" that can dynamically connect the various applications. For the open modularity concept to work, the software selected needs to be preferably based on freeware or relatively inexpensive packages. The criteria that need to be used in selecting the software is complex but needs to include the following [16, 17]:

- Cost and usability; (training time needed, applications it can handle etc)

- Expandability to new versions
- Help and support available from the supplier and/ or user groups
- User friendliness of the HCI
- Complexity of the environment that can be handled; (1D, 2D, 3D, static or dynamic situations)
- Simulation aspects provided (robot(s), task(s), environment(s))

The chosen software must be able to provide the ability to design and simulate each of the generic modules needed. In addition it needs to allow three design and simulation functions to be performed, namely allow for the design of robots, the working environments to be set up and it must allow designers to study tasks that need to be performed. Many software applications have been developed and are available. A list of over 100 titles has been identified and the selection criteria for choosing the best for specific cases are presented in [17]. The software selection process identified software for the System and Modular Levels. Of these, the ones presented in the following list were considered to be the most appropriate for the Modular Level Design.

- Visual Nastran for specific limb motion and 3D Design
- Yobotics; Similar to visual Nastran including Biomedical Robot design, with built-in control system
- Darwin2K a dynamic simulation and automated design synthesis package for robotics

4. Colonoscopy Robot Inspection Case Study

The case study discusses the modular design of a mobile robot for colonoscopy [1, 12]. It introduces and proposes design environments for mobile climbing robots on irregular and/ or sleek terrains. There are two general categories of climbing scenarios depending on the application; the first includes all kinds of terrain vertical or horizontal, rough or smooth, while the second arises in medical applications where wet and dry, rough and smooth, and rigid and pressure-deformed surfaces need to be addressed.

The ability to handle these types of surfaces and various robot kinematic and dynamic mechanics needed to be simulated in the appropriate internal body environments. In order to visualize the problem(s) and invent new solutions, they an be studied in the simulated set up before risking having to build anything. Applications other than medical ones can be simulated by various software packages that provide 3D design and simulation engines together with CAD capabilities for kinematic analysis. As a result of the research carried out on the software packages available, packages like Visual Nastran, Darwin2K and Yobotics were discussed to be the most appropriate. Visual Nastran is felt to be most appropriate and will be used in this case study for specific limb motion as it gives a variety of joint types and motions with various properties to match different design and simulation concepts. Player Stage (system level tool) is another powerful simulation studio which includes mapping, localization both in 2D and 3D with on board camera views, path planning and 3D world design and all these make it a useful robot design tool. Darwin2k is a combination of the player-stage system lelev software but with module level design capabilities. It has not being tested yet but demos have been run and it is also considered and believed to be used.

The CLAWAR design approach to start with formulating the system level requirements and breaking these into appropriate modules level sub-designs is followed here. Regarding any application, the system level design should come before the modular level design. The details of how a system level approach can be realised is discussed in [17] and is used in this case study in order to continue to the modular level design. The design aspects for the modular design are focussed upon here.

For the application of colonoscopy, 3D environments are vital for realising effective robot designs. All mobile robots in any environment need to sense, make some decisions and move themselves or move something. This is also the case for robot colonoscopy systems. By completing the system level design first it is possible to produce a list of the necessary modules from considering the task requirements regarding the specific application. The colonoscopy robot should be biocompatible and able to navigate within the restrictive confines of the large intestine. The main constraint apart from the biocompatibility and strength for locomotion is that it must be compact and able to stabilize itself while navigating. In other words a mechanism must allow the robot to either grip or stick to the walls around it without rupturing them and be able to manoeuvre freely.

Having this in mind the System Level Tasks are listed below:

Working Environment: *The Inner Human Body:*

a. 3D, Intestine (Colon) – tight confined environment

b. Pressure-deformed surfaces

c. Danger of rupturing walls

d. Sleek/ smooth walls

e. Liquid flow might be present

Task to be carried out

a. Semi-autonomous in navigation

b. Allow medical examination

c. Provide visual feedback to doctors

d. Perform medical procedures (remove tumours)

Performance metrics

a. Safe: biocompatible

b. Effective: locomotion, inspection, treatment

c. Compact

d. Reliable

<u>Operation</u>

a. Remote

b. Semi-autonomous

This way of approach highlights the need for different software for the environment design and simulation. Firstly any possible CAD or robot design and simulation software could work well for applications involving rigid pipes in petrochemical plant situations [6, 14], but the introduction of soft tissue tubes in biomedical situation such as in colonoscopy exposes the need for a new type of software where pressure-deformed surfaces and tissue dynamics are included [8]. Pressure-deformed surfaces lead to the need to alter the locomotion methods. End effectors or sensors (grippers/manipulators, pressure sensors, camera, blood flow sensors, temperature sensors, distance; and collision avoidance detection sensors) or mechanics are the system level design requirements. The “pressure-deformed” characteristic of the colonoscopy environment leads to the formulation of the following list of requirements:

- Safety: need for umbilical to retrieve device
- Soft surface dynamics: need for special contact mechanisms
- Mechanics: “wet” bioactive environment.
- Motion planning: Movement without damaging delicate intestine wall
- Umbilical: useful for power, communication to/ from device

Simulating soft surface dynamics and in general the whole internal-to-body environments such as those of the colon is not widely available. Legged locomotion on soft ground has been considered [5] but to the authors knowledge there is no software suitable for simulation of internal body environments. In view of this rigid pipes have been used to simulate the colonoscopy application. Hence in the simulations the pressure at the contact points assuming rigid pipes is monitored and controlled so that it does not exceed the threshold for causing rupturing of the colon wall.

CLAWAR’s generic methodology uses four basic modules to interact via six variables (power, mechanics, data bus, analogue signals, digital signals and the environment) to integrate with each other to allowing applications specific solutions to be formed. These main basic modules that are needed comprise sensors actuators, power supplies, computing hardware, software, communication devices and the materials, [17 - 21]. In order to realise and implement the open design methodology we need to determine a good way of integrating the modules by having open protocols for the interfacing allowing the modules to be seen as "black boxes". In order to do this we introduce the Common Module Block shown in Fig. 3 where the input and output variables can be specified.

Module Title:		
INPUT	Extra Details	OUTPUT
Power:		Power:
Analogue:		Analogue:
Digital:		Digital:
Databus:		Databus:
Environment:		Environment:
Mechanics:		Mechanics:

Fig. 3. The Module Block representing the inputs and outputs categorized into the six interaction-space variables.

The "input variables" simply state the input requirements of the module while the "output variables" categorize what is being output for the six interfacing space variable. When a few modules have been designed so that they match and can be integrated, it is hoped that a mature and robust methodology and set of standards can be determined that will be acceptable to the CLAWAR and wider robot communities. Having as reference the subsumption architecture which Brooks established [22], we can extend it to have the emphasis on modular components and how they can integrate together; the concept is shown more clearly in fig. 4.

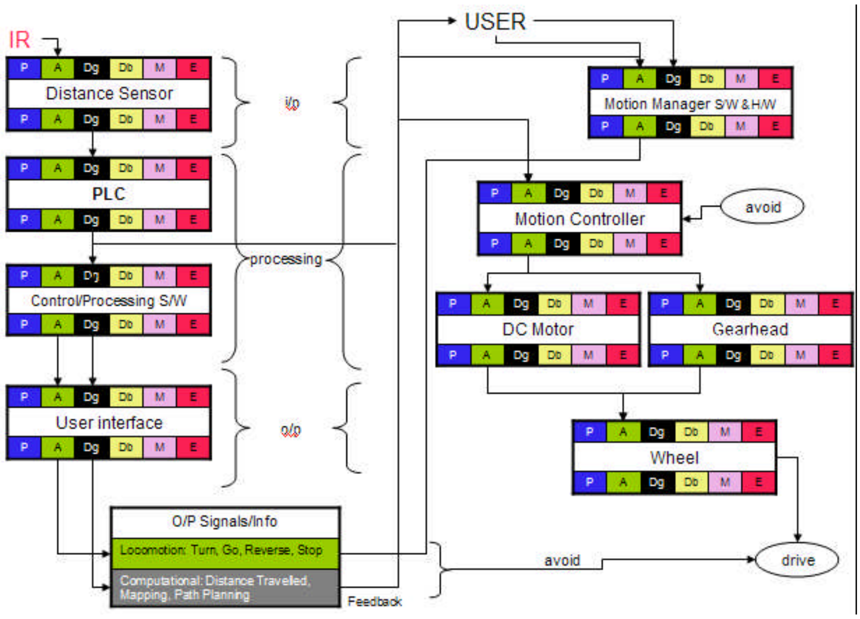

Fig. 4. Module Architecture

The basic concept is that the modular approach can be followed up to build up specific sense-processing-output paths within the robot's actions to build up the design attributes to meet the system level task requirements. In this way the designer can grow the modular level design using an appropriate toolset that allows him to test the inter-connectivity of the modules while assessing if the overall system can perform the required tasks.

5. Conclusions

The paper has presented a generic method to support the CLAWAR open modular approach. Appropriate software tools (using freeware software as much as possible) are needed to allow designs to be put together and tested in a virtual way before actual building is initiated. In this way modular components can selected from a data base to realise the functionalities needed and speed up the process. The software most suitable has been investigated for this purpose and Player-Stage is felt to be the most appropriate. A case study involving colonoscopy has been presented to highlight the details of the generic methodology and this has highlighted the shortcomings in the available tools, namely no software appears to be available for simulating soft tissue applications.

References
[1-22]

1. A. Menciassi, J.H.P., S. Lee, S. Gorini, P. Dario, Jong-Oh Park. *Robotic Solutions and Mechanisms for a Semi-Autonomous Endoscope*. in *IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*. 2002. EPFL, Lausanne, Switzerland2.
2. B.L. Luk, D.S.C., A.A. Collie, N.D. Hewer, S. Chen. *Intelligent Legged Climbing Service Robot For Remote Inspection And Maintenance In Hazardous Environments*. in *8th IEEE Conference on Mechatrinics and Machine Vision in Practice*. 2001. Konk Kong.
3. Bose, A.K., *Modular Robotics*, Department of Mechanical Engineering, Rashtreeya Vidyalaya College of Engineering: Bangalore. p. 1-16.
4. David J. Bruemmer, D.D.D., Mark D. McKay, Matthew O. Anderson, *Dynamic-Autonomy for Remote Robotic Sensor Deployment*, The Human-System Simulation Laboratory, Idaho National Engineering and Environmental Laboratory.
5. Iikka Leppanen, Sami Salmi and Aarne Halme, *Workpartner - Hut Automation's new Hybrid Walking Machine*.IMSRI, Clawar98, 1998.
6. Hisanori Amano, K.O., Tzyh-Jong Tarn. *Development of vertically moving robot with gripping handrails for fire-fighting*. in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2001. Maui, Hawaii, USA.
7. Jesse A. Reichler, F.D., *Dynamics Simulation and Controller Interfacing for Legged Robots.* The International Journal of Robotics Research. **19**(01): p. 41-57.

8. L. France, J.L., A. Angelidis, P. Meseure, M.-P. Cani, F. Faure, C. Chaillou, *A Layered Model of a Virtual Human Intestine for Surgery Simulation.* Elsevier Science, 2004: p. 1-20.
9. Luis E. Navarro-Serment, R.G., Christiaan J.J. Paredis, Pradeep K. Khosla, *Modularity in small distributed robots*, Institute for Complex Engineered Systems, The Robotics Institute, and Department of Electrical and Computer Engineering, Carnegie Mellon University: Pittsburgh, Pennsylvania 15213. p. 1-10.
10. Mark A. Minor, R.M., *Under-Actuated Kinematic Structures For Miniature Climbing Robots.* ASME Journal of Mechanical Design, 2002.
11. Michael Beetz, T.B., *Autonomous Environment and Task Adaptation for Robotic Agents*, University of Bonn, Dept. of Computer Science III: Bonn, Germany.
12. P. M. Y. GOH, K.K., *Microrobotics in surgical practice.* British Journal of Surgery, 1997. **84**: p. 2–4.
13. P. S. Schenker, P.P., B. Balaram, K. S. Ali, A. Trebi-Ollennu, T. L. Huntsberger, H. Aghazarian, B. A. Kennedy and E. T. Baumgartner, K. Iagnemma, A. Rzepniewski, and S. Dubowsky, P. C. Leger and D. Apostolopoulos, G. T. McKee, *Reconfigurable robots for all terrain exploration*, Jet Propulsion Laboratory, Massachusetts Institute of Technology, Carnegie Mellon University, University of Reading (UK). p. 1-15.
14. R. Aracil, R.S., O. Reinoso, *Parallel robots for autonomous climbing along tubular structures.* Robotics and Autonomous Systems, © 2002 Elsevier Science B.V., 2002/3. **42**: p. 125–134.
15. S Galt, B.L.L., D.S. Cooke, A.A. Collie, *A tele-operated semi-intelligent climbing robot for nuclear applications*, IEEE, Editor. 1997, Department of Electrical and Electronic Engineering, University of Portsmouth - Portsmouth Technology Consultants Ltd. p. 118-123.
16. Williams, L.D.a.G., *Evaluating and selecting simulating software using the analytic hierarchy process.* Integrated Manufacturing Systems, 1994. **5**(1): p. 23-32.
17. Yiannis Gatsoulis, I.Chochlidakis., Gurvinder S. Virk. *Design Toolset for Realising Robotic Systems*. in *CLAWAR 2004*. 2004. Madrid.
18. G.S. Virk, *CLAWAR Modularity for Robotic Systems* The International Journal of Robotics research, Vol.22, No. 3-4, MArch-April 2003, pp. 265-277, Sage Publications
19. G.S. Virk, *Technical Task 1. Modularity for Clawar machines - Specifications and possible Solutions* Clawar Network, 1999
20. G.S. Virk, *Task 1 Report on "Modularity for Clawar Machines - Specification and Possible Solutions"*, Year 1 Report to the EC for CLAWAR, EC Contract Number BRRT-CT97-5030, 1999
21. P. Maly, G.S. Virk, I. Kagalidis, D. Howard, A. Vitko and L. Jurisica, *Modular Design For Robotic Systems*
22. Rodney, A. Brooks, *A Robust Layered Control for a mobile Robot*, A.I. Memo 864, September 1985, Massachusets Institute of technology