

# Modelling of Modular Robot Configurations Using Graph Theory

José Baca, Ariadna Yerpès, Manuel Ferre, Juan A. Escalera, and Rafael Aracil

Universidad Politécnica de Madrid,  
Jose Gutierrez Abascal, 2, 28006 Madrid, Spain  
{jbbaca,ayerpes,jescalera,aracil}@etsii.upm.es, m.ferre@upm.es

**Abstract.** Modular robots are systems that can change its geometry or configuration when connecting more modules or when rearranging them in a different manner to perform a variety of tasks. Graph theory can be used to describe modular robots configurations, hence the possibility to determine the flexibility of the robot to move from one point to another. When the robot's configurations are represented in a mathematical way, forward kinematics can be obtained.

**Keywords:** Modular Robots, Graph Theory, Configurations for Displacement.

## 1 Introduction

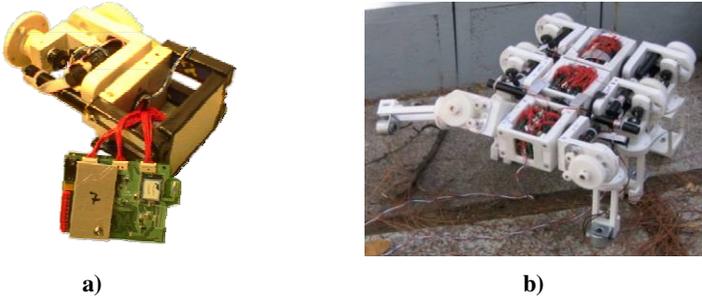
Now in days, different modular robots designs have been proposed to give a solution to varied fields like versatility, adaptability, robustness, costs, etc. It is possible to find module robot designs from 1 degree of freedom (DOF) [1], 2 DOF [2], [3] and 3 DOF [4], [5]. Each system shows diverse relevant advantages between them, a few of them are reconfigurable and others are autoconfigurable systems, but something in common is that they can form new robot structures when combining modules. When attempting to obtain the model of a defined robot is just a matter of following a systematic procedure. With a defined robot, the number of degrees of freedom, length's links, masses and geometry are normally well defined and constant, facilitating its modeling. On the contrary, modular robots are more complex to model, due to its capability of increasing the DOF, and links when adding modules to the structure. The kinematics of the robot will be defined by the number of modules and the way they are connected. Therefore it is necessary to use a method to easily represent the structure's configuration.

## 2 RobMAT Architecture

The desire of any modular system is that the whole is greater than the sum of its parts. The main advantage of using modular robots lies in the possibility of constructing different kinematics chains by combining modules.

RobMAT's system architecture is divided into modules, molecules and colonies. A module (figure 1 (a)) is the base component of the system and it has movement and

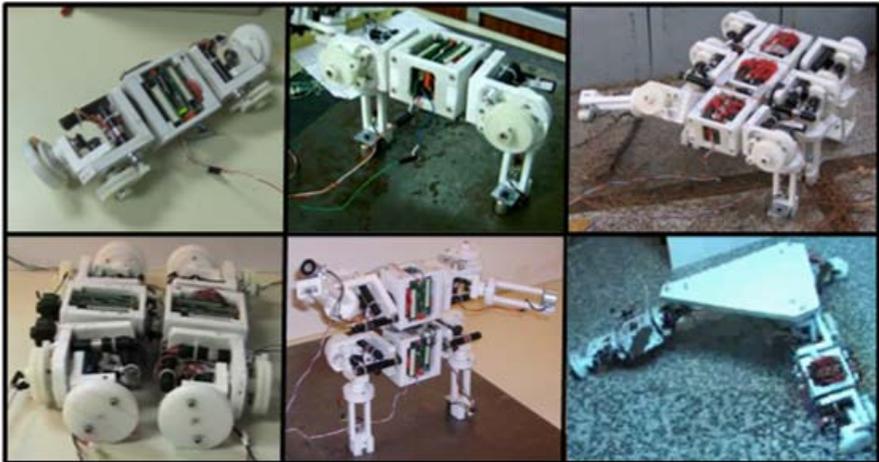
communication capacity. A **molecule** (figure 1 (b)) is an autonomous entity made out of n-modules connected together ( $n > 1$ ). The molecule holds higher capabilities of movement, communication and manipulation than the module. Various molecules cooperating in the area to fulfill a task is defined as **colony**.



**Fig. 1.** a) Base component of the RobMAT system. b) Six module configuration.

The system has three communication channels in order to properly transmit commands to the whole system and to be able to work in a cooperative mode. These communication channels are the Intra-Module Communication, the Inter-Modules Communication and the Video Communication.

The first type is in charge of the communication between modules belonging to the same molecule and it uses a CAN bus [6]. The Inter-Modules Communication is in charge of communication between molecules or between a molecule and the control station and it is based on Bluetooth. Finally, the Video Communication is used to send radio frequency video signals from the two mini cameras that can be attached to one of the molecules and that would be used to control the system in a remote working environment using teleoperation.



**Fig. 2.** By combining modules, distinct modular robot configurations can be formed, creating many ways of displacement and manipulation

The used module had an actuator with 3 rotational degrees of freedom with its axes intersecting at one point creating a wrist-like joint. The design of the module allows accessories to be attached on one end of the module and the connector used to enable reconfigurations is made out of an electromagnet (male connector) and a metallic plate (female connector), each part embedded in one of the molecules making possible to form different modular robot configurations as is shown in figure 2.

### 3 Describing Modular Robot Configurations

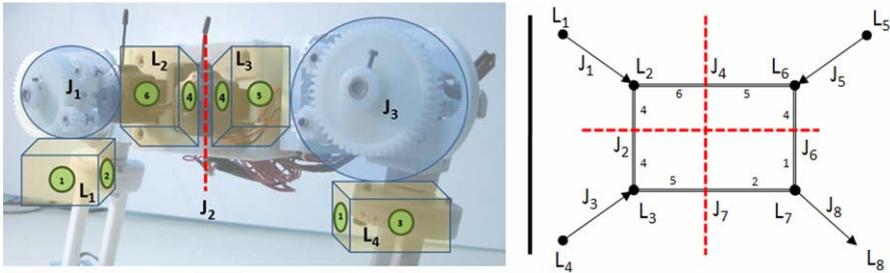
The flexibility of modular robots provides enormous adaptation capacity and the ability to perform a number of tasks. However, it complicates the robot's kinematics and dynamic modeling [7]. The changing configuration of molecules means that, unlike other robots, modeling in advance is not possible. An algorithm is therefore required to automatically generate the model for any molecule configuration during the execution of each step of a task.

#### 3.1 Graph Theory

Kinematics graphs are a widely-used tool for the theoretical analysis of mechanisms [8]. These graphs entail vertices and edges (or lines), which represent robot links and joints respectively. All elements of a mechanism belong to one of these categories, so any mechanism can be represented by a graph. Graph Theory [9] offers many theoretical tools to analyze graphs, and applied to mechanisms, specifically talking about the graphs of mechanisms; it provides the opportunity to analyze them in a more abstract manner. A graph is a mathematical object that captures the notion of connection.

#### 3.2 Graph Theory into Modular Robots

A modular system can be represented by its corresponding graph, showing the relationship existing among different modules. The application of Graph Theory to modular robots requires the use of specific graph elements. This degree of specialization is achieved by labeling graph vertices and edges in such a way that there are as many labels as different types of robot links and joints. If we consider RobMAT as a homogenous robot, i.e. without considering the tools it can handle, it can be seen in general one type of **link** (each prism (p) next to the spherical joint), two types of **joint** (connector (c) and spherical (s) joint), and another element to be taken into account when representing module chains is the linking point between modules, which is called the **port**. Basically, a connector can have more than one place or port to join with other connector. Generally, a number codifies each port and is located in the graph close to the corresponding element where the linking is produced. The figure 3 (a) shows representative graph elements over each robot module. Each module is composed of two links, one joint and its ports (e.g. module one is composed of  $L_1$ ,  $L_2$ , and  $J_1$  and module two is composed of  $L_3$ ,  $L_4$  and  $J_3$ ), but when combining modules a joint between them is created due to the port at the link section (e.g.  $J_2$  is created when both modules are connected at port 4 in both cases). The figure 3 (b) shows a four module configuration graph and it can be noticed the joints created at each module union (i.e.  $J_2$ ,  $J_4$ ,  $J_6$ , and  $J_7$ ) and its ports are showed.



**Fig. 3.** a) Representative graph elements over each RobMAT module. b) 4-module configuration graph.

All this information is set forth in the Assembly Incidence Matrix AIM [10], so that it can be easily included in algorithms. The AIM is a  $(N+1) \times (M+1)$  matrix with  $N$  vertices ( $v$ ) and  $M$  edges ( $e$ ). This matrix is formed by giving to each entry  $a_{ij}$  the number of the port that joins  $v_i$  and  $e_j$ , or 0 when no linking appears. The extra column  $(M+1)$  indicates the link type, while the extra row  $(N+1)$  shows the joint type. Figure 3 (a) shows the RobMAT module with its ports and their assigned numbers. It can be seen that module has six possible connection ports with other modules. To clear up some final concepts, Figure 3 (b) shows the graph of the 4-module RobMAT molecule, where each link is named  $L_i$  with  $i=1,2,\dots,8$  and each joint is named  $J_i$  with  $i=1,2,\dots,8$ . To distinguish between the two kinds of joints, spherical joints are represented as a single-line edge and connectors as a double-line edge. Also, the spherical joint edge is directed (from  $-1$  to  $1$ ) to show the order of its three axes. The arrow indicates where the end-effector link is located. Also, the number at both ends sides of each connector indicates the port of connection between the links. The graph therefore contains enough information to describe the modular robot assembly.

The AIM for the Figure 3 (b) graph is:

$$AIM = \begin{matrix} & \begin{matrix} J_1 & J_2 & J_3 & J_4 & J_5 & J_6 & J_7 & J_8 \end{matrix} & \begin{matrix} | \\ | \\ | \\ | \\ | \\ | \\ | \\ | \end{matrix} \\ \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 4 & 0 & 6 & 0 & 0 & 0 & 0 \\ 0 & 4 & 1 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 1 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ s & c & s & c & s & c & s & 0 \end{pmatrix} & \begin{matrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \end{matrix} \end{matrix}$$

### 3.3 Modular Robot Configurations for Displacement Determined by the Complexity of the Graphs

Many configurations can be built for displacement, but just a few of them are efficient for the task. By analyzing the graph (links and joints structure), it can be determined the complexity for the robot to move from one point to another. An example of this situation can be noticed when comparing the six modules configuration from figure 4 with the six modules configuration from figure 5. Although the same number of

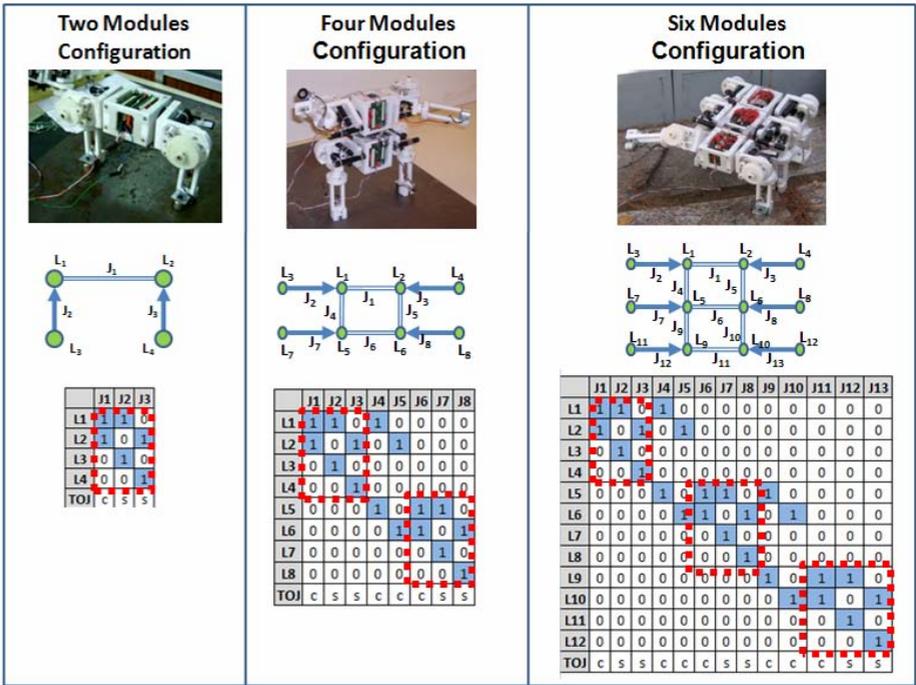


Fig. 4. Each configuration can be represented by graphs and with this a mathematical way to describe the structure generated

modules are used, and thanks to an additional platform (accessory) as common connection between each pair of modules in figure 5, the flexibility of the robot to execute displacement is higher when less links and joints appears at the graph.

Therefore, during real applications, it is important to seek out for equilibrium between the complexity of the task and the configuration of the modular robot, For example. the use of two molecules which have simple graphs due to its configuration (e.g. four modules configuration at figure 4) to carry on with the task, instead of using one molecule compound of many modules with high pull-push torque as a result, but forming a complex graph full of several links and joints (e.g. six modules configuration at figure 4).

When analyzing modular robot configurations in a simple way, it can be seen repeatable assembly patterns. This repeatability will depend undoubtedly when using the same configuration of a pair of modules to form chains, as is shown in figure 4. By this observation, it is achievable to describe the actual configuration in a simplified manner. The figure 4 shows different configurations formed by two, four, and six modules chains. It can be detected that the matrix (pattern) obtained by the two modules graph is repeated at four and six module matrix due to the use of a second and third same pair of modules configuration, the only difference is the connection between each set.

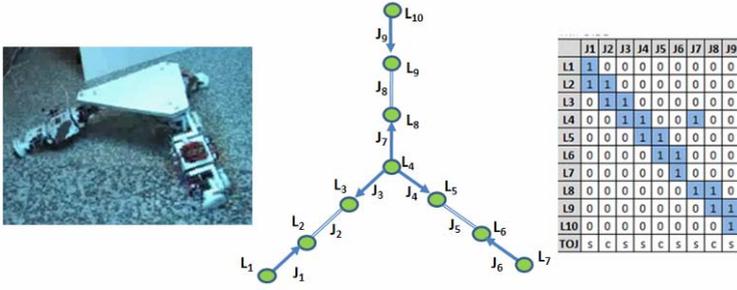


Fig. 5. While smaller the complexity of the graph, the highest flexibility of the robot to move

### 4 Forward Kinematics

Once a configuration can be described in a mathematical formulation the corresponding kinematical model can be obtained. All joints in the RobMAT are rotational; nevertheless modules can have any kind of joint. For this reason, it is helpful to apply Screw Theory [11]. This allows treating prismatic and rotational joints in the same expressions without specific changes. Also, almost any joint can be decomposed as a combination of those basic ones. A forward kinematics equation of an open chain robot can be uniformly expressed as a product of exponentials (POE) [12]. Using POE and Graph Theory, forward kinematics of tree-type modular robots can be obtained. The following equation is an example that solves RobMAT module direct kinematics (Figure 3). Let  $q_1, q_2$  and  $q_3$  be the generalized coordinates of the 3-DoF RobMAT module. The corresponding POE formula is given as follows:

$$H_e^e(q_1, q_2, q_3) = e^{\hat{S}_1 q_1} e^{\hat{S}_2 q_2} e^{\hat{S}_3 q_3} H_e^0(0)$$

where:

$$\hat{S}_1 = \begin{pmatrix} w_1 = (0,0,-1) \\ v_1 = (0,0,0) \end{pmatrix}$$

$$\hat{S}_2 = \begin{pmatrix} w_2 = (0,-1,0) \\ v_2 = \left(-\frac{L}{2}, 0,0\right) \end{pmatrix}$$

$$\hat{S}_3 = \begin{pmatrix} w_3 = (1,0,0) \\ v_3 = \left(0,-\frac{L}{2},0\right) \end{pmatrix}$$

$$H_e^0 = \begin{pmatrix} 0 & 0 & 1 & \frac{L}{2} \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & \frac{L}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$(\hat{S}_i)$  is a screw and is expressed in the base frame (0). Furthermore, all coordinates are expressed in the base frame, and also in the zero position which is a fixed position.  $H_e^0(0)$  is the well-known homogenous transformation between the end-effector frame (e) and the base frame at zero position. Once the module kinematics model has been determined, the kinematics of molecules can be obtained by compounding several module kinematics. Hence, an arbitrary module is designated as root, and position/orientation is propagated from this root to every end-side module through the modules in the molecule. In this way, the forward kinematical model is obtained.

To automate this process, it is important to know how modules are connected to each other. Depending on where the module port is attached, orientation and/or position changes will or will not be required. According to the port number in both links, a fixed homogeneous transformation can be obtained that provides the reference frames of neighbour modules.

For instance, the following described forward kinematics model of link  $L_8$  (figure 3 (b)). The root link is  $L_1$  and one possible path from  $L_1$  to  $L_8$ , which is obtained from the graph or AIM, would be  $\{L_1, J_1, L_2, J_2, L_3, J_7, L_7, J_8, L_8\}$ . Then, for each vertex on the path, a homogeneous transformation is added as shown in the next expression.

$$H = H_e^0 \cdot H_4^e \cdot D \cdot (H_4^e)^{-1} \cdot H_5^e \cdot D \cdot (H_2^0)^{-1} \cdot H_e^0$$

where:

$$H_1^0 = H_4^e = I_4$$

$$H_2^0 = H_6^e = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -w/2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{array}{l} \text{Transformations} \\ \text{between} \\ \text{reference frames} \\ \text{and connectors} \end{array}$$

$$H_3^0 = H_5^e = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & w/2 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$D = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{Docking matrix}$$

Thus, by evaluating all paths from the root to the pending modules of each molecule, the full forward kinematics model is solved. This model is very important in order to obtain molecule workspace and to determine its functionality.

## 5 Conclusions

The possibility of modelling modular robot configurations can be achieved by graph theory. Its study generates a mathematical object so it can be manipulated to perform complex analyzes, like forward kinematics. Analyzing the graph of each corresponding robot configuration and comparing them with tested robot displacements a conclusion show up. The more complex the graph is (i.e. high number of links and joints), the robot flexibility to move from one point to another decreases, due to the reduced workspace of the DOF when having many modules so close. Although it is possible to represent any modular robot structure with graphs, just some configurations will be useful for real applications. Therefore, the search for equilibrium between complexity of the task and modular robot configuration must be achieved.

## References

1. Yim, M., Duff, D., Roufas, K.D.: Walk on the Wild Side. *IEEE Robotics & Automation Magazine* 9(4), 49–53 (2002)
2. Murata, S., Yoshida, E., Kamimura, A., Kurokawa, H., Tomita, K.Y., Kokaji, S.: M-TRAN: Self-Reconfigurable Modular Robot System. *IEEE/ASME Trans. Mechatronics* 7(4), 431–441 (2002)
3. Castano, A., Behar, A., Will, P.M.: The CONRO Modules for Reconfigurable Robots. *IEEE/ASME Trans. Mechatronics* 7, 403–409 (2002)
4. Shen, W.-M., Krivokon, M., Chiu, H., Everist, J., Rubenstein, M., Venkatesh, J.: Multi-mode Locomotion for Reconfigurable Robots. *Autonomous Robots* 20(2), 165–177 (2006)
5. Escalera, J.A., Ferre, M., Aracil, R., Baca, J.: ROBMAT: Teleoperation of a Modular Robot for Collaborative Manipulation. In: *Proc. 11th International Conference, Knowledge-Based Intelligent Information and Engineering Systems, Part 2*, pp. 1204–1213 (September 2007)
6. Crespo, A., Baca, J., Yerpes, A., Ferre, M., Aracil, R., Escalera, J.A.: CAN Application in Modular Systems. In: *12th International CAN Conference, Barcelona, Spain*, pp. 01-1 – 01-7 (2008)
7. Chen, I.-M., Yang, G.: Automatic generation of dynamics for modular robots with hybrid geometry. In: *Proceedings of 1997 IEEE International Conference on Robotics and Automation*, April 20-25, vol. 3, pp. 2288–2293 (1997)
8. Freudenstein, F., Dobrjanskyj, L.: On a Theory for the Type Synthesis of Mechanisms. In: *Proc. 11th Int. Congress of Applied Mechanics, Munich*, pp. 420–428 (1964)
9. Gross, J.L., Yellen, J.: *Graph Theory and Its applications*. Chapman & Hall/CRC (2006)
10. Chen, I.-M., Yang, G., Kang, I.-G.: Numerical inverse kinematics for modular reconfigurable robots. *Robotic Systems J.* 14, 213–225 (1999)
11. Ball, R.: *A Treatise on the Theory of Screws*. Cambridge University Press, Cambridge (1900)
12. Brockett, R.: Robotic Manipulators and the Product of Exponential Formula. In: *Mathematical Theory of Networks and Systems*, pp. 120–129 (1984)