Isomorphic Gait Execution in Homogeneous Modular Robots

Michael Park, Sachin Chitta, and Mark Yim

University of Pennsylvania
20 August 2006

Motivation

• Wish to control a Modular Robotic structure composed of uniquely labeled modules, regardless of labeling order.

• Need for an organized structure to exploit the versatility of Modular Robots, with a library of known configurations with corresponding gaits.

• Bridging the interface between Posable Programming and Embedded isomorphic control.

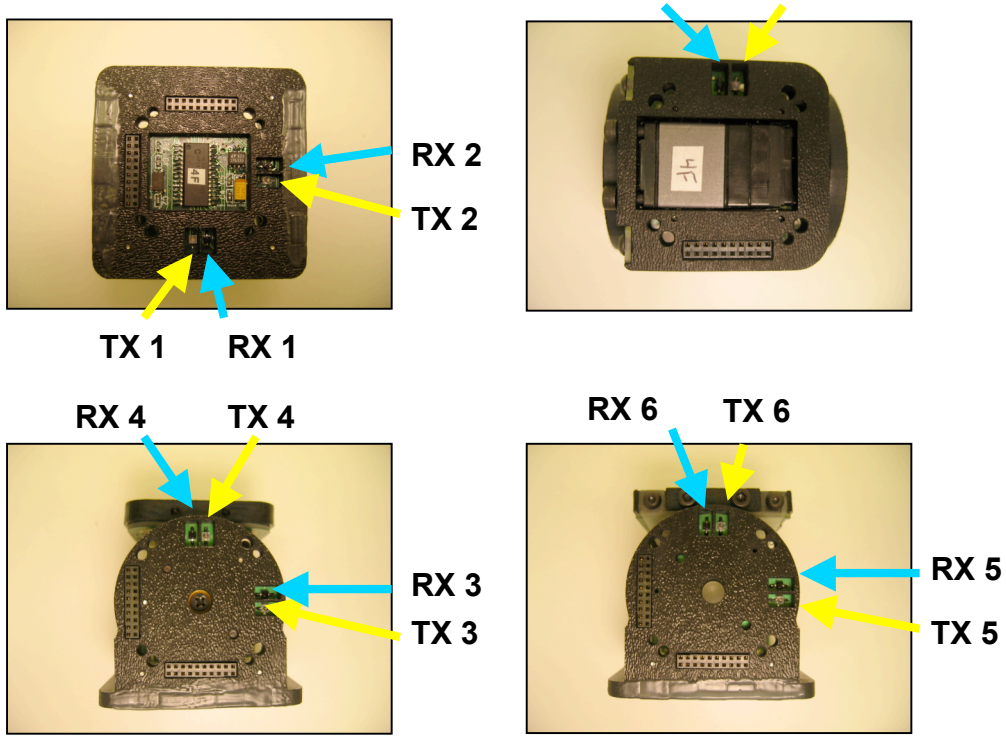
Penn

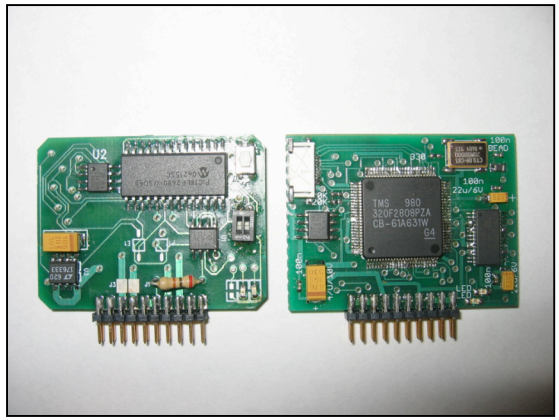
Hardware

•Reconfigurable single-degree-of-freedom unit modular robot.

•Seven Infrared transmitters & receivers per module; connections between modules detected and transmitted over a bus network.

•Single Controller sub-module reads configuration data, searches for a match in database, outputs actuator values when match is found.



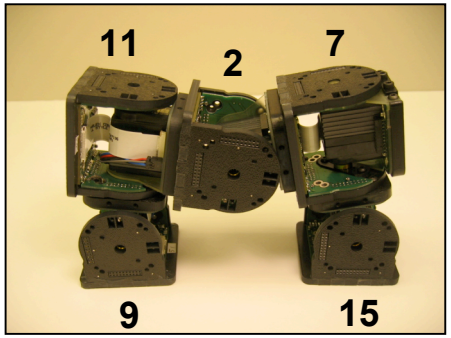


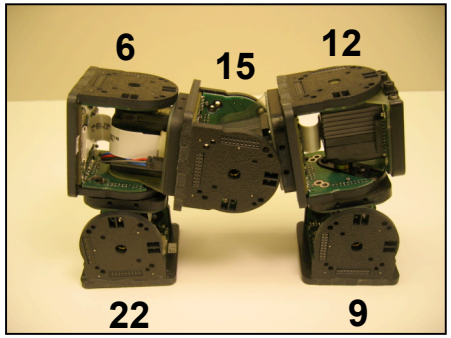
Current Controller

Controller In Development

Penn

Different Labelings, Same Shape: Graph Isomorphism Problem





Port Adjacency Matrices

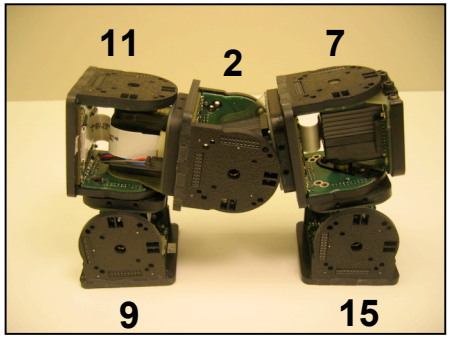
$$A = \begin{array}{c} \\ 2 \\ 7 \\ 9 \\ 11 \\ 15 \end{array} \begin{array}{c} \begin{array}{ccccc} 2 & 7 & 9 & 11 & 15 \end{array} \\ \begin{bmatrix} 0 & 7 & 0 & 2 & 0 \\ 2 & 0 & 0 & 0 & 4 \\ 0 & 0 & 0 & 7 & 0 \\ 7 & 0 & 6 & 0 & 0 \\ 0 & 7 & 0 & 0 & 0 \end{bmatrix} \end{array} \qquad A' = \begin{array}{c} \\ 6 \\ 9 \\ 12 \\ 15 \\ 22 \end{array} \begin{array}{c} \begin{array}{ccccc} 6 & 9 & 12 & 15 & 22 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 7 & 6 \\ 0 & 0 & 7 & 0 & 0 \\ 0 & 4 & 0 & 2 & 0 \\ 2 & 0 & 7 & 0 & 0 \\ 7 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

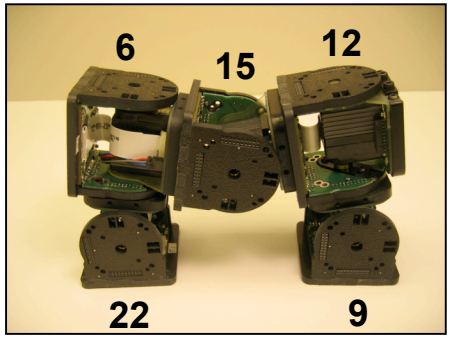
Isomorphic Structures have the same Characteristic Polynomial

$$\begin{aligned} \mathrm{Det}\,(\mathbf{A} - \lambda \mathbf{I}) &= \mathrm{Det}\,(\mathbf{A}' - \lambda \mathbf{I}) \\ &= -\lambda^5 + 98\,\lambda^3 - 2156\,\lambda \end{aligned}$$

Penn

Different Labelings, Same Shape: Graph Isomorphism Problem





Physical, Logical ID Mappings, & Permutation of Logical IDs

Physical IDs of **A**
$(2 \quad 7 \quad 9 \quad 11 \quad 15)$
Logical IDs
$(L1 \quad L2 \quad L3 \quad L4 \quad L5)$

$$\pi_{A \to A'} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 3 & 5 & 1 & 2 \end{pmatrix}$$

Physical IDs of **A′**
$(15 \quad 12 \quad 22 \quad 6 \quad 9)$
Logical IDs
$(L4 \quad L3 \quad L5 \quad L1 \quad L2)$

Corresponding Output (angles) for Lurching Gait

	2	7	9	11	15
step 1	4	2	0	2	1
step 2	15	3	-10	2	22
step 3	42	3	29	2	23
step 4	-24	3	29	2	23
step 5	10	3	29	3	-34
step 6	-19	3	25	3	-44

	15	12	22	6	9
step 1	4	2	0	2	1
step 2	15	3	-10	2	22
step 3	42	3	29	2	23
step 4	-24	3	29	2	23
step 5	10	3	29	3	-34
step 6	-19	3	25	3	-44

Penn

Configuration Library & Gaits

- Controller sub-module recognizes ~50 configurations out of ~10^7 possible configurations (up to 7 modules, each with 7 ports).
- 7 module configuration and execution can be implemented on a very simple PIC processor.
- Configurations and corresponding gaits are inputted manually or created and uploaded with a GUI.

Penn

Posable Programming GUI

- Windows GUI records Port Adjacency Matrix and corresponding angular values for a given configuration.
- Data is uploaded to the Controller sub-module where the configuration is integrated into the Library of configurations and corresponding gaits.
- Permutations of Library configurations are compared with Port Adjacency Matrices acquired from the IR transmitters and receivers.

Penn

Future Work

- Implementation for ~16 modules. Off-board computing interface using Nauty/Bluetooth, hierarchical control using communicating clusters of sub-modules, more capable processor.
- Gait multiplicity for configurations, possibly treating each feedback signal as an additional element in the Port Adjacency Matrix
- Integration with user-input controls.

Penn