Reconfigurable Robot
What is a Reconfigurable Robot?

- Self-reconfiguring modular robots are autonomous kinematic machines with variable morphology.

- They are able to deliberately change their own shape by rearranging the connectivity of their parts, in order to adapt to new circumstances, perform new tasks, or recover from damage.

- For example:
  - worm shape, narrow pipe
  - spider shape, uneven terrain
  - ball shape, flat terrain
Motivation and Inspiration

- **Functional advantage:**
  - More robust and adaptive than conventional systems
    - Hardware reconfigurability

- **Economic advantage:**
  - Can lower overall robot cost by making a range of complex machines out of a few types of mass-produced modules
Potential Applications

- **Space exploration**
  - Requires long-term self-sustaining robotic ecology that can handle unforeseen situations and may require self repair

- Space missions are highly volume and mass constrained. Sending a robot that can reconfigure to achieve many tasks is better than sending many special purpose robots
Taxonomy of Architectures

- **Lattice vs. Chain** based on the geometric arrangement of their unit

- **Lattice architectures**
  - Units are arranged and connected in some regular, space-filling 3-D pattern, such as a cubical grid
  - Control and motion are executed in parallel
  - Simpler computational representation and more easily to scale to complex systems

- **Chain/tree architectures**
  - Units are connected in a string or tree topology
  - The underline architecture is serial
  - They can reach any point in space, and are more versatile but more computationally difficult to represent and analyze
Different Types of Reconfigurable Robots

- Minimal-capability modules:
  - Chain-type links (rolling track, spider, snake, etc.)

- Lattices, matrices (for stair climbing, object support, etc.)

- Connectable robots

- Modular, plug-and-play components
Chain-Type Example: PolyBot/Polypod

- Work by Yim, Xerox
- PolyBot/Polypod:
  - Bi-unit modular robot
  - Consists of exactly two types of modules repeated many times
  - Repetition makes manufacturing easier and cheaper
  - Dynamic reconfigurability enables robot to be versatile and reconfigurable
Movies of PolyBot
(Mark Yim, Xerox PARC)

Self-reconfiguration: loop, snake, spider
Fence climbing

http://www2.parc.com/spl/projects/modrobots/chain/index.html
Chain-Type Example: CONRO

- Work by Castano, et al., USC/ISI
- **Goal**: development of taskable reconfigurable robot
- **Composition**: identical modules programmable for altering its topology
- **Base topology**: simply connected, as in a snake
- **More complex topology**: includes appendages
- **Each module**: CPU, memory, battery, micro-motor plus sensors
- **Major challenges**: packaging, power, cooling, programming and program control

http://www.isi.edu/robots/conro/
Movies of CONRO
(USC Information Science Institute)

From snake to a rolling track
Reconfiguration
Lattice-Type Example: Robotic Molecule

- Work by Rus, Dartmouth
- The Molecule
  - 4-degree-of-freedom
  - Can aggregate with other identical modules to form 3D dynamic structures

http://groups.csail.mit.edu/drl/wiki/index.php/Main_Page
Objective: Develop reconfigurable system for reaching variety of hard-to-reach areas

http://staff.aist.go.jp/e.yoshida/smaunit/index-e.htm
Connectable Robots Example
Hirose et al., Tokyo

- Connectable robots for reconfiguration, linked mobility
- Challenging new area with potentially high payoff
  - Broad cross-domain application
    - Subsets of modules may operate independently or combine as circumstance necessitates
    - Cooperative object transport
    - Reconnaissance
  - Manually-linked mobility has been demonstrated, but not autonomous reconfiguration
- Possibility to increase the capabilities of existing multi-robot systems
  - Navigate previously inaccessible terrain
Modular Plug-and-Play: Millibot
Khosla et al, CMU

- Objective: Develop small (5-10cm) modular robots capable of surveillance and reconnaissance
- Manually reconfigurable robots
Today’s Readings

- “Hormone-Inspired Adaptive Communication and Distributed Control for CONRO Self-Reconfigurable Robots”, by Shen et al.
  - Presented by Michael Ortiz

- “Crystalline Robots: Self-reconfiguration with compressible unit modules”, by Rus and Vona.
  - Presented by Joseph Paredes
Introduction

- **Self-reconfigurable robots**
  - Are made of autonomous modules that can connect to each other to form different configurations
  - Each module is autonomous
  - Modules must collaborate and synchronize their actions in order to accomplish desired global effects
Two Basic Problems Addressed

- How modules communicate with each other when connections between them may be changed dynamically and unexpectedly?

- How these physically coupled modules collaborate, in a distributed manner, their local actions to accomplish global effects?

- Module coordination needs to be:
  - distributed: to host a large number of modules
  - dynamic: to deal with changes
  - asynchronous: to compensate the lack of global clocks
  - scalable: to support shape-changing
  - reliable: to recover from failure
Communication

- A self-reconfigurable robot can be viewed as a network of nodes that can change their connections dynamically.
- Dynamically changing the topology requires continually determining the route.
- There’s a need to rediscover connection at the module level.
- An adaptive communication (AC) protocol is designed for all modules to discover and monitor their local topology.
Coordination

- The modules should coordinate their actions by their locations in the current configuration, not by their names.

- A control message may also require concerted actions.

- **Hormone-based control**
  - This concept is used among biological cells for both communication and control.
  - A single “hormone” signal can propagate through the entire network, yet cause different modules to react differently based on the local information.
Properties of a Hormone Signal

- A hormone signal is similar to a content-based message:
  - It has no specific destination
  - It propagates through the network
  - It may have a life time
  - It may trigger different actions for different receiver
  - Hormone signals may be modified during its propagation
CONRO Robot Module

- Each module is autonomous
- It’s equipped with:
  - a microcontroller
  - two motors
    - pitch (up and down)
    - yaw (left and right)
  - two batteries
  - four connectors
    - front, left, right, back
  - four pairs of IR emitter/sensor for communication
CONRO Modules

- When a male connector and a female connector are joined together, it forms an active link.
- The connected modules are called neighbors.
- CONRO modules communicate with each other through active links.
- Each connector has an IR transmitter and receiver.
- Two pairs form a bidirectional local communication link.

9x4 connectors forming a hexapod
8 active links
A CONTRO Robot as a Network

- Each node is a module
- Each node has finite connectors
- Each edge is an active link
- The topology of the network may change dynamically and active links may appear or disappear
- Nodes can only communicate through active links
- Nodes do not know the network size nor have unique IDs
Local Topology Types and Example

### TABLE 1: THE LOCAL TOPOLOGY TYPES OF CONRO MODULES

<table>
<thead>
<tr>
<th>Connected to other modules</th>
<th>This Module</th>
<th>This Module</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td>Connected to other modules</td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>f</td>
</tr>
</tbody>
</table>

![Diagram of CONRO modules and their connections](image)

IR Receiver: 🟢 IR Transmitter: ⭕ Yaw motor: ⬤
(Pitch motors not shown)
The AC Protocol

- The main procedure is a loop of receiving and sending “probe” hormones between neighbors
  - A probe is a special type of hormone that is used for discovering and monitoring local topology
  - After one exchange of probes between two neighbors, both sides will update their current active links
  - Every module probes its neighbors every cycle

- If the network is a acyclic graph, the AC guarantees that every nonprobe message will be propagated to every module once and only once
  - When a module receives a hormone, it will send it to all its active links except for the one from which the hormone is received
Hormone-Inspired Distributed Control

- A distributed protocol that is identity free but supports a module to select its actions based on its location in the network.
  - Since hormones can trigger different actions at different site.
  - Every module can rediscover its local topology.
- We will use hormone messages here.
6-Module CONTRO Snake

- T1-T16-T16-T16-T16-T2
- To move forward, each module’s pitch motor goes through a series of positions
- Problems with a control table:
  - it does not deal with dynamics
Rulebase for the Move

**Shifting pattern**
- A module $m$ at step $t$ is the action to be performed by the module $(m-1)$ at step $(t+1)$
- A hormone can be used to propagate through the snake and allow each module to inform its immediate neighbor what action it has selected so the neighbor can select the appropriate action and continue the hormone propagation
Compared to Static Gait Table

**Advantages:**
- it supports online reconfiguration and is robust to a class of shape alterations
  - A snake cut into two segments
- Scalability, it works no matter how many modules are in the snake
- Efficiency
  - Only one hormone is used to coordinate the behavior
More Movies of CONRO

- Snake configuration moves with caterpillar gait
- Snake sidewinder on pebble
- Snake reconfiguration
- Rolling track
Key Issues Addressed in CONRO

- Adaptive communication
- Collaboration between physically coupled modules

Solution must be:
- Dynamic: to deal with changes in network topology
- Asynchronous: to compensate for lack of global clocks
- Scalable: to support shape-changing
- Reliable: to recover from local damages
Hormone Inspiration in CONRO

“Hormone” signal:
- Propagates through entire network
- Causes different modules to react differently based upon:
  - Local receptors
  - Topology connections
  - State information
- No specific destination
- May have a lifetime
Communication

- Communication makes use of topology definitions.

- **Local topology** determined by how CONRO modules are inter-connected:
  - Back, Front, Right, Left

- Depending on connections, topology type given a name:
  - T0, T1, ..., T31

- Modules continually verify their current local topology through “probing” hormones.
Adaptive and Distributed Control Protocol

- Use of Rulebase for various types of moves
- Modules decide what to do based on:
  - Local topology
  - Local state information (e.g., timer, motor and sensor states)
  - Received hormone messages
- **Rulebase inputs:**
  - Module type
  - Local timer
  - Received hormone data
- **Rulebase outputs:**
  - Action for module to perform
  - Hormone for module to send
Various CONRO Rulebase Developed

- Rulebase for:
  - Caterpillar move
  - Legged walk
  - Rolling track
  - Etc.

- Rulebases developed manually; unclear how to automate this