Proceedings of 2004 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 28 - October 2, 2004, Sendai, Japan

Sensor-Based Distributed Control for Chain-Typed Self-Reconfiguration

Kenneth Payne, Benham Salemi, Peter Will, Wei-Min Shen
Information Sciences Institute
University of Southern California
Marina del Rey, CA, United States
kpayne@usc.edu, {salemi, will, shen}@isi.edu

Abstract— **This paper describes two contributions for chain-typed self-reconfigurable robots: a very illustrative self-reconfiguration task changing from "I" shape to "T" shape, and a sensor-based distributed control method for automatic planning and execution of self-reconfiguration. In the "I-to-T" task, a snake robot is to reconfigure itself into a tripod by docking the tail to a target module in the body, releasing a portion of the connected mass as a new leg, and switching to a new gait automatically. We first accomplished this task using predetermined instructions for individual modules without considering sensor inputs. We then developed a sensor-based approach using our hormone-inspired distributed control to allow the robot to dynamically accept the point of connection at run-time, align the tail and the target using sensors, and select appropriate actions based on modules' location in the configuration. Compared to the standard inverse kinematics, this new control approach is sensor-based and can endure the limited computational resources and uncertainties in the connections. It can be applied to self-reconfigurations that are not designed by the programmers but triggered by the environment.**

Keywords- chain-type, digital hormone; sensing; CONRO; closed loop; reconfiguration; self reconfiguration; snake robot; triped; tripod; SMA;

I. INTRODUCTION

Dynamic planning and execution of chain-typed self-reconfiguration is a very challenging problem. This is due to the diversity in tasks, the lack of benchmark testing, the limitation of mechanical and computational resources on-board, and the inherited uncertainties in the reconfigurable joints. Traditional methods such as inverse kinematics require a powerful processor in each module or a centralized off-board controller [1]. Distributed control methods often require pre-scripted instructions for individual modules to compensate for the lack of dynamic planning and execution abilities at run-time [2].

This paper makes two contributions for chain-typed self-reconfiguration. It first defines a very illustrative shape-changing task called "I-to-T". In this task, a snake robot is to reconfigure itself into a tripod by docking the tail to a target module in the body, releasing a portion of the connected mass as a new leg, and switching to a new gait automatically. We first accomplished this task using predetermined instructions for individual modules without considering sensor inputs. The second contribution of this paper is a closed-loop sensor-based approach using our hormone-inspired distributed control to allow the robot to dynamically accept the point of connection at run-time, align the tail and the target using sensors, and select appropriate actions based on modules' location in the configuration. Compared to a standard inverse kinematics method, this new control approach is sensor-based and can endure the limited computational resources and uncertainties in the connections. It can be applied to self-reconfigurations that are independent to the length of the snake and the location of the target connection point. Such a method can be generalized for self-reconfiguration tasks that are not designed by the programmers but triggered by the environment.

There is extensive literature on both docking with multiple degrees of freedom, as well as docking with reconfigurable systems. In the case of CONRO, like many other modular systems, docking must be accomplished despite the accumulation of tolerances across multiple connections and joints of actuation. Atkins and Murphy [7] proposed the requirements for just such an operation, across incrementing tolerances. Nilsson [8] designed a connector capable of self-alignment. Roufas et al. [9] created a connection system that aligned through the use of infrared sensors, but was very complex at six degrees of freedom. Fukuda and Kawauchi [10] and Murata et. al. [11] all explored docking and connecting with robotic arms. Robotic Molecules [12-13] were able to dock successfully through the use of magnetic and electro-magnetic docks and then with grippers to assist in alignment and positive connection. Most of these involved the use of a single system of modules, changing the overall structure through reconfiguration. Rubenstein, et al. [14] was able to dock two independent CONRO systems through the use of infrared sensors for alignment.

II. EARLY SOLUTIONS TO THE "I-TO-T" TASK

Peter Will first suggested the "I-to-T" task as a standard test for chain-typed self-reconfigurable robots. His initial vision was to reconfigure a snake robot into a 'T'-shaped robot that is capable of a crawling gait. Although deceptively simple, this task would demonstrate self-reconfiguration requiring a small number of both docking and undocking actions. The resulting changes in the topology of the robot would be sufficient to require a new mode of locomotion.

There are several earlier scripted solutions to this problem. In July 2002, Støy and Blynel[1] visited the CONRO project and started implementing a scripted solution using 'role-based control' [16] but did not complete the task due to the time. Salemi later completed their implementation and successfully demonstrated the first "I-to-T" behavior using hand-coded instructions for the modules and a single, predetermined target connection point. The parallel execution of actions was implemented through a pre-specified delay mechanism. However, the control was open-loop and no sensor information was used during run-time.

In 2003, Salemi and Shen implemented another open-loop solution using the hormone-inspired distributed control [3] so that the reconfiguration sequence can be triggered at any module in the snake. A synchronization mechanism was developed to control the parallel and serial action sequences, and actions could have various durations. However, the instructions for individual modules were still hand-coded, no sensor feedback was used for alignment and docking, and the connection point was still predetermined.

Advanced from the above earlier solutions, this paper describes a new implementation of the hormone-inspired distributed control in which the robot does not need to know beforehand the size of the overall configuration and the location of the connection point, nor does each module need a prescribed angle value. At runtime, modules' roles are determined as a function of the number of modules between the tail and the target module, and movements and alignment are guided by the sensor feedback from the environment.

III. THE USE OF DIGITAL HORMONES IN CONRO

The focus of this section is the use of digital hormones on the self-reconfigurable robot CONRO. Before addressing the experiment proper, first we will describe the CONRO robot and the fundamentals of digital hormone control.

A. The CONRO Self-Reconfigurable Robot

A CONRO system consists of multiple self-reconfigurable robotic modules connected in a network structure. An individual module, shown in Fig. 1, has two servo-driven axes (pitch and yaw), four docking connectors (one female, three male) and a BASIC Stamp 2SX controller. Communication is point-to-point via infrared transmitter-receiver pairs on the four dock faces. Power is supplied through a two-conductor 6V DC tether. Modules can only communicate to their immediate neighbors using serial communication protocol via infrared transmitters and receivers.

A typical module is 10.8cm long (face to face), 4.4cm wide, and 4.5cm high (measured without the cables). There are two variations of CONRO module: East- and West-type. Because of the orientation of the main control board, some modules have more East movement about the yaw axis, and some more toward the West. However, all modules are calibrated about the centerline along the length of the module.

There are also two variations of the docking connector. The newer loop-lock version and the older pin-and-latch style are described in detail in [15]. This experiment uses the pin-and-latch version. Essentially, the connector uses a pair of 0.5cm diameter pins that extend 0.6cm into the female latch connector. These pins have grooves cut into them, which allows the latch to seat, thereby holding the pin in place. The latch itself is a spring-loaded, normally closed lever of the same thickness as the pin grooves. To open it, a momentary current is given to a shape memory alloy (SMA) which contracts enough to move the latch.

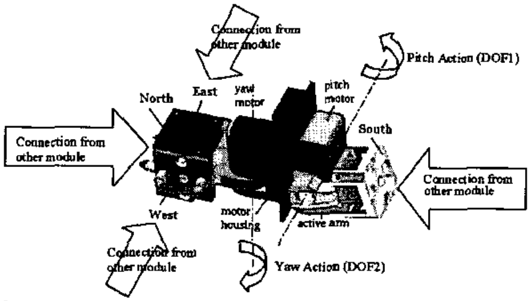

Figure 1. A typical CONRO module, with latch-style connectors.

There are several hardware limitations that directly affect the success of self-reconfiguration tasks. For example, the off-the-shelf infrared receivers saturate before actual docking has been achieved, and also have a tendency to peak at about ± 5° off-center. The friction on the table magnifies the error tolerances of the yaw joints (± 2°) and individual connections (± 1°). The modules themselves absorb some of the torque applied during docking.

B. Digital Hormone Control, (DHC)

Digital hormones, described at length in [3], are a method of control that is ideal for homogenous networks of robotic modules. The modules in the network need no IDs, nor do they need to know anything about the network except their immediate position in the topology. A single digital hormone message will generally remain constant throughout its propagation, and each module will act according to its content and the module's position in the local topology. As a digital hormone message is released into the neighboring modules, it may take one of several different courses: (1) It may be passed along to the next module's neighbors. (2) It may be inhibited, and therefore not propagated. (3) It may be modified and released again. (4) It may be stopped, and an entirely new and different hormone message released in its place.

The best way to illustrate digital hormones is from the point of view of each individual module. These modules have no IDs, so they differentiate from one another through their local topological connections, (and also through the exception of temporary counter messages, explained later). For example, in a snake configuration, if a module's

---

[1] Prof. Kasper Stoy visited the CONRO project from the University of Southern Denmark from June 2001 to August 2002, Jesper Blynel a Ph.D student from The University of Lausanne visited in the summer of 2002.

connectors are all idle except that the north connector is connected to the south connector of another module, then this module is the "tail" of the snake. This is acceptable, as it mimics a real biological system. The hormone is released, but it is up to each recipient whether to ignore it or to perform a response.

Fig. 2 illustrates the process within the module. As a hormone is received, a set of rules (analogous to "receptors" in biological cells) is applied to it. The outcome of the rules is based on the local topology, the current state of the module, sensor values, and the received message itself. The rule set effectively isolates various kinds of modules involved in a certain task, while the others simply maintain the system and pass messages. For example, if the specified action is intended for "head" modules, (modules docked only to the female connection), then a certain action is taken, whereas all other modules will be performing a different action because their local topology will differ regardless of how similar their states and sensor readings may be.

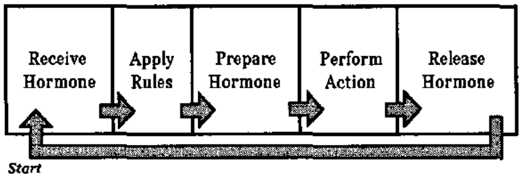

Figure 2. The process of digital hormone control. Note that the action is always performed *before* an outgoing hormone can be released.

A typical application for DHC on a modular robot is for locomotion. A *go* message is released into the system, and each module acts according to its location in the local topology as appropriate to that message. Usually this is the "set-up" message that tells all the modules to wait for a *start* message before beginning locomotion. Eventually, a dynamically elected module [6] eligible to start the movement will reply with the start message, and each module will in turn propagate it and act accordingly. In this simple example, a hormone is propagated only to those modules that have not received it before (it is inhibited by those modules that have already received it). In a quadruped, those modules acting as spines, head, and tail may sway back and forth, while the left and right leg modules may move in a cyclic motion to gain ground. If a module's position changed in the topology (e.g., a leg exchanged for a spine, for example), the module would detect the new topology and select an appropriate action for the new position. For example, the old leg would start acting like a spine, and the spine a leg.

Another application for DHC is conducting self-reconfiguration. This however, is not so straightforward as locomotion. Many more hormones are released in response to each other during reconfiguration as opposed to one or two being released in locomotion. During reconfiguration, some modules may be searching for others that may be lighting a beacon, signaling across open space, or simply maintaining a physical connection in the network. The use of digital hormones for such tasks is relatively new. The main advantage is that this biologically inspired method can use the limited computation resources onboard efficiently: (the BASIC Stamp II-SX used in CONRO modules has only 32 words of variable space and 8K memory). Such tight resources allow no room for inverse kinematics or calculating precise positions. Instead, we use a sensor-based approach, taking into account the precision limitations of both the servo positions and the backlash in the docking connectors. This is both easier to implement on the controller, as well as closer to the biological feedback system used in moving an arm in nature.

IV. RECONFIGURATION FROM SNAKE TO TRIPOD

To reconfigure from a snake into a tripod, the operator cues a module selected at random along the snake, which then becomes the target point of connection. If the touched module (the "target") is far enough from the tail, the system bends the tail and target into close proximity. Next, the tail and target take turns seeking and aligning with each other, while they are slowly brought closer together [14,17]. The sensors eventually read a value greater than the threshold required for docking, indicating that the tail and target are not only aligned, but close enough to dock. At this point, the snake bends sharply in the middle and presses the tail into the target dock site, and then disconnects part of the loop to release a leg onto the side of the tripod. Finally, the tripod swims away with a starfish-inspired gait. Fig. 3 illustrates the process as viewed from the overall system. Fig. 4 defines the different modules in the system. Movies of this self-reconfiguration process can be found at http://www.isi.edu/robots.

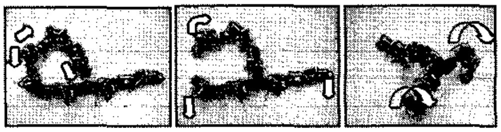

Figure 3. System starts as a snake, the tail aligns with and docks to the body at the target module, detaches a leg, and then "swims" off.

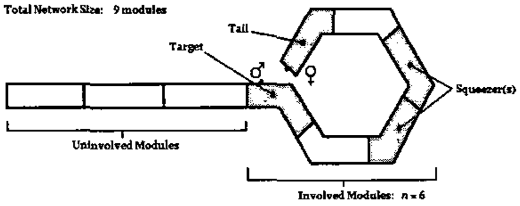

Figure 4. A typical reconfiguration will have two kinds of modules: Involved and uninvolved. Of the involved modules, there are the tail, target, and normal modules, some of which classified as *squeezers*.

There are several actions that are used in the snake-to-tripod experiment. These are, for the most part, entered sequentially, with the exception that the *Target Seeking* and *Tail Seeking* behaviors will cycle for multiple iterations.

These actions are triggered primarily through the reception of hormones of the same name, and are listed in Table I.

TABLE I. ACTIONS/HORMONES INVOLVED IN SNAKE-TO-TRIPOD

Action/ Hormone	Description
Idle	Performs the initial action before reconfiguration is triggered.
Counting	Passes a counter variable south to the tail module.
Wait for Final Count	Module has either counted once, or is North of a module that counted already.
Final Count	Sent by the tail, this returns the amount of modules involved in the reconfiguration, including the tail and target.
Final Count Received	Sent by the head module, this message lets the tail know that all modules have received the final tally.
Go to Start Position	This hormone is released by the tail. As it passes each module, another counter is propagated with it. This one decrements its counter with each module it passes.
Target Seeking	This is released by the target when it seeks. It will make one pass and then move internally into *Light Beacon*. The hormone is released after the pass.
Tail Seeking	This is released by the tail when it seeks. It will make one pass and then move internally into *Light Beacon*. The hormone is released after the pass.
Docking	This hormone is released when either the tail or target has determined that a dock is possible.
Walk Away	Once the leg has connected, and the target module detects it, this hormone is released, allowing each module to move in a sinusoidal oscillation, moving out from the center. This hormone also tells the squeezer closest to the target to release its South SMA latch, detaching the group.
Light Beacon	This is a private action and is never released externally as a hormone. It allows the module to spend a full action cycle alighting the infrared beacon.
Error	This hormone is released whenever a fault is detected in the system. The system is restarted.

It is worth noting that the above behaviors can be rearranged for various tasks besides attaching a leg to a snake. With minimal modifications to the above hormones, they can be used to attach a leg to a different location, remove legs, dock two independent systems together, and break a larger robot into smaller mobile units. To accomplish any of these other tasks, one would simply change the rule set, leaving the static behaviors above relatively unmodified.

A. The Mechanics of Reconfiguration

At the signal from the operator (simulating a condition in the environment), the target module releases a counter hormone that increments as it passes through the network. When the tail module receives the hormone, it checks to see if there are enough modules between the target and the tail to ensure a successful docking. For CONRO, it takes at least 5 modules to form a docking loop, and this condition is shown in Equation (1), where n is the number of all modules, including the target and tail, involved in the loop.

$$n:\ n > 5 \tag{1}$$

If this condition is not satisfied, then the tail would propagate an *Error* hormone to reset the snake. Otherwise, the snake will bend its body so that the connectors of the tail and the target will face each other in a close proximity. Since the target module is selected at run-time, the modules compute their bending angles based on the geometry of a regular polygon. In the resulting position, the tail can usually "see" the target connector's beacon signal through its infrared sensors during a pass, and vice-versa. This position can be easily achieved by bending all the modules with an angle that is dynamically determined by the number of modules involved in the chain.

Recall from elementary geometry of regular polygons that the internal angle of a regular polygon is in Equation (2), where θ_i is the interior angle in radians, and n is the number of sides.

$$\theta_i:\ \theta_i = (n-2)\pi / n \tag{2}$$

If the target is the head, the tail were docking to the North dock of the target, and all modules achieved their allotted orientation in this arrangement, then they would create a regular polygon of n sides, with side lengths of 10.8cm (a full module length) between the yaw servomotors (see the left diagram in Fig. 5). However, in this experiment, the tail is to dock to a side connector (East or West) and must maintain orientation with the target, thus the target dockside must point to the tail, and vice-versa.

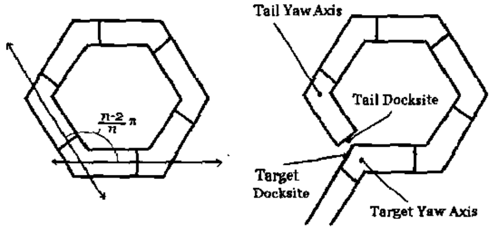

Figure 5. The start position is based very closely on the regular polygon that a head-to-tail dock would create. In this fashion, the intended dock is in very close proximity (i.e., the sensors can "see" each other).

We chose to make the target bend the opposite direction with the same interior angle. This is simple to calculate, and begins with both faces parallel but not yet aligned. As shown in the right diagram in Fig 5, as the line from the tail to the target yaw motor would be perpendicular to the surface of the target dock, and already by definition perpendicular to the tail dock face.

Lastly, to start with the tail beacon already pointed in the highest-probable direction of the target receiver, we must add 0.24 radians to θ_i to determine the internal angle for the tail module start position. This is because the length of a module and the distance from the yaw joint to the dock face are both fixed in CONRO modules, and is exactly 0.24 radians in the above starting configuration.

Once all the modules are in position, the tail module emits its infrared beacon while the target module begins a "receiver" sweep across its yaw axis. Once the target finds the highest response on that sweep, it moves to that position and lights its beacon. The tail then begins its sweep to find the newly aligned target module. After that, modules in the middle of the snake, called *squeezers*, pinch their yaw joints in the direction of the dock (clockwise for East-side docking and counter-clockwise for West-side docking), and this will bring the tail and the target closer in distance. This sweep/search/squeeze process is repeated until their receiver values exceed a predetermined docking threshold value. This means the two docking faces are now close enough and their connectors are parallel and aligned. When this happens, the tail module releases a *Docking* hormone.

During the propagation of the *Docking* hormone, all involved modules bend their yaw servos inwards, while the tail vibrates (to create dynamic lubrication) and the target rotates back and forth about its yaw axis servo (to increase the compliance for the final docking). As the sides are forced together, the tail and target correct any deviation from the docking alignment through the compliant motion of the target yaw servo and the dynamic lubrication of the tail vibration. This stage determines when the connectors are fully latched on each other, and can be detected by the receiver values of infrared signals.

After the connection is established, the target releases a *Walk Away* hormone that tells the squeezer to undock from its South connector, cutting the loop in two, releasing the leg. Also as the hormone propagates to all modules, each in turn begins a simple tripod "starfish gait".

B. Rules

To dynamically choreograph a set of digital hormones, all incoming messages are evaluated by a fixed set of rules for each kind of task. The rules can be likened to the next-state logic of a finite state machine. They compare the incoming hormone-message content with the current state or action of the module, as well as other pertinent data: local topology, sensor values, internal counters, message history, and the like.

TABLE II. THE RULES FOR THE TAIL MODULE (GROUP 1)

Next State	**Current State**	***Hormone-In*, Other Factors**
Final Count	*Idle*	*Counting* AND $n > 5$
Error	*Idle*	*Counting* AND $n \leq 5$
Go To Start Position	*Final Count*	*Final Count Received*
Tail Seeking	*Go To Start Position*	*Target Seeking*
Light Beacon[a]	*Tail Seeking*	*Tail Seeking* sent
Tail Seeking	*Light Beacon*	*Target Seeking*
Docking[b]	*Tail Seeking*	IR > *Docking* Threshold

a. Executed internally, no hormone released. b. At this point, tail changes to an uninvolved module (waits for *Walk Away* hormone).

The state of each module consists primarily of the current action as well as the module's place in the immediate topology. There are four main "groups" of modules in the snake-to-tripod experiment: The tail (open South dock), the target (point of attachment for the new leg), those involved with the operation (between the tail and target), and those not involved (North of the target). The rules for these four groups are listed in Tables II, III, IV, and V. Notice that since the target module is determined at run-time, the last two groups must be determined dynamically while parsing the *Go to Start Position* message. This dynamic role determination is unique and made possible by the distributed and flexible nature of the hormone-inspired control. Since modules in this system do not require unique global IDs, actions must be performed in response to hormones and local topology. For this reason, hormone messages may record path information (such as the number of hops in propagation) to collect knowledge of the surrounding topology. For example, during the *Counting* behavior, an incrementing counter is propagated along the network, augmented each time it moves toward the South, (headed for the module with a free South connector). This is used to determine the number of modules between the target and the tail.

TABLE III. THE RULES FOR THE TARGET MODULE (GROUP 2)

Next State	**Current State**	***Hormone-In*, Other Factors**
Wait for Final Count	*Idle*	*Counting*
Final Count Received	*Wait for Final Count*	*Final Count*
Target Seeking	*Final Count Received*	*Go to Start Position*
Light Beacon[a]	*Target Seeking*	*Target Seeking* sent
Target Seeking	*Light Beacon*	*Tail Seeking*
Docking	*Target Seeking*	IR > *Docking* Threshold
Walk Away	*Docking*	New Link Added

a. Executed internally, no hormone released.

Those modules that are in the third group also have two more roles: *squeezer* and *non-squeezer*. These roles are determined as "counter" hormones propagate through the system, and by Equation (3) during the *Tail Seeking* behavior. Note that during *Docking*, all involved modules will squeeze, whereas there will be only a few squeezers during *Tail Seeking*.

$$\text{Squeezer}_{Tail\ Seek}\text{: count} = (n/2)\%1 \text{ OR } ((n+1)/2)\%1 \tag{3}$$

During the *Tail Seeking* action, a decrementing counter is propagated along the network. If the decrementing counter is greater than 0, then the module is considered to be "involved" directly with the reconfiguration process, and the count is further applied to (3) to determine if the module should squeeze, or simply release the hormone further.

Since the involved modules are passively involved in the process (except for the squeezers, which participate, but also do not alter any messages), the rules are somewhat lengthy, but easy to follow. For the most part, they simply do what their neighbor is doing, watching for their turn to squeeze (if applicable).

To determine the "uninvolved" modules (Group 4), the decrementing counter is set to the number of modules between the target and tail, and released from the tail during the *Go to Start Position* action. If the counter is greater than 0, then the module is involved and the hormone is released again with a smaller counter. If the counter is less than 1, then the hormone is inhibited and the module becomes uninvolved (dormant) until either the *Walk Away* or *Error* hormone is received.

TABLE IV. THE RULES FOR INVOLVED MODULES (GROUP 3)

Next State	Current State	***Hormone-In*, Other Factors**
Counting	*Idle*	*Counting* (from North)
Wait for Final Count	*Counting*	*Counting* (from South) [a]
Final Count	*Counting*	*Final Count* [b]
Final Count	*Wait for Final Count*	Final Count
Final Count Received	*Final Count*	*Final Count Received*
Go to Start Position	*Final Count Received*	*Go to Start Position* AND (count > 0)
Target Seeking	*Go to Start Position*	*Target Seeking*
Tail Seeking AND squeeze	*Target Seeking*	*Tail Seeking* AND (count = (n/2) % 1 OR count = ((n+1)/2) % 1)
Tail Seeking (No squeeze)	*Target Seeking*	*Tail Seeking* AND (count ≠ (n/2) % 1 OR count ≠ ((n+1)/2) % 1)
Target Seeking	*Tail Seeking*	*Target Seeking*
Docking AND squeeze	*Target Seeking*	*Docking* AND count = (n/2) % 1
Docking (No squeeze)	*Target Seeking*	*Docking* AND count ≠ (n/2) % 1
Docking AND squeeze	*Tail Seeking*	*Docking* AND count = (n/2) % 1
Docking (No squeeze)	*Tail Seeking*	*Docking* AND count ≠ (n/2) % 1
Walk Away	*Docking*	*Walk Away*

a. Never happens to uninvolved modules. See text. b. Occurs in the module directly adjacent to the tail module.

The rules for the uninvolved modules are listed in Table V. Note that there will never be an occasion where they will receive a *Counting* hormone from their North neighbors, (receiving one from the North means that the module is between the target and tail, and the message travels North-to-South). For both involved and uninvolved modules, receiving a *Counting* hormone through the South connector means releasing the *Wait for Final Count* hormone.

TABLE V. THE RULES FOR THE UNINVOLVED MODULES (GROUP 4)

Next State	Current State	***Hormone-In*, Other Factors**
Wait for Final Count	*Idle*	*Counting* (from North)
Final Count	*Wait for Final Count*	*Final Count*
Final Count Received	*Final Count*	*Final Count Received*
Final Count Received	*Final Count Received*	*Go to Start Position* AND count < 1
Final Count Received	*Final Count Received*	*Docking*
Walk Away	*Final Count Received*	*Walk Away*

Lastly, for all modules, there are certain universal rules. To deal with conflicting tasks, (task negotiation), all tasks were given a priority level in the order listed above lowest-to-highest. With this in mind, the general rules are listed in Table VI.

TABLE VI. GENERAL RULES FOR SNAKE-TO-TRIPOD SELF-RECONFIGURATION

Action Taken	General Condition
Inhibit Received Hormone	Current State > Received Hormone AND Current State ≠ *Light Beacon* AND Received Hormone ∉ {*Seekers*}
Propagate Hormone	Current State ≤ Received Hormone OR Current State = *Light Beacon* OR Received Hormone ∉ {*Seekers*}
Release New Hormone	Current State was changed AND Current State ≠ *Light Beacon*
Inhibit Received Hormone	If no other rules were applied

Lastly, if an inappropriate message was received, a beacon was lost, or a module dropped a message, the system propagates the *Error* hormone, which halts and restarts the experiment. Note that all modules have the same set of rules listed in the Table I-VI, it is their location in the system that determines their roles in different stages of this self-reconfiguration.

V. EXPERIMENTAL RESULTS

The "I-to-T" task was tested with a snake configuration where a human operator signals an arbitrary module to be the target. During the alignment process, the tail always came within 4cm face-to-face with the target dock. Of those times that the tail and dock aligned, the pins would go straight into the holes 90% of the time during the final squeeze. A majority of the attempts would continue to dock even with the modules starting up to 4° out of alignment. Movies of this self-reconfiguration process can be found at http://www.isi.edu/robots.

In these experiments, we have varied the length of the snake from 6 to 10, the target position from one end to the other. Among this series of 20 consecutive experiments, no cases ever returned the Error state, and all cases started the docking processes, and out of which 18 docking experiments (90%) succeeded. The two failed docking experiments were due to noise in sensors and friction on the table. The control logic is correct in all cases. We are working to improve the reliability of the hardware to increase the rate of success in self-reconfiguration.

In these experiments, we devised several solutions that allowed the system better success in docking. We improved the docking process by increasing the compliance of the docking pairs through the addition of small magnets to the docking faces. This did not affect the alignment process, as the magnets are not strong enough to deflect the passes of the scanning phase. This still required the pins to be aligned with the latch holes. In addition, we reset the brightness range in the infrared detectors by adding peephole blinders to them. This corrected the off-center alignment, and also reduced the amount of light that could enter from the distances involved in docking. However, it still allowed for point-blank reading, for when the modules would communicate once connected.

VI. CONCLUSION

As shown, the application of digital hormones to control self-reconfiguration is possible and works effectively. Also, because it does not rely on complex inverse kinematics, it can be ported to very simple controllers with a minimum amount of variable space and program memory.

For future work, we will extend the approach to include the higher-level control sequence to perform leg-attachment several times to produce quadrupeds, hexapods and other combinations. Likewise, one could use another approach to "deconstruct" a pedal arrangement into the snake configuration. This is possible because the hormones from Table I can be rearranged with a different set of rules to allow the completion of other tasks. Another area of work would be letting a condition in the environment select the target module. We will modify the modules to have appropriate sensors for detecting environmental triggers.

ACKNOWLEDGMENT

We are grateful that this research is sponsored by AFOSR under award numbers F49620-01-1-0020 and F49620-01-1-0441. We also want to thank other members in the Polymorphic Robotics Laboratory for their generous intellectual support and discussions.

REFERENCES

[1] Yim, M., D.G. Duff, K.D. Roufas. "PolyBot: A modular reconfigurable robot," in Proceedings of the IEEE International Conference on Robotics and Automation. 2000.

[2] Yoshida, E., Murata, S., Kamimura, A., Tomita, K., Kurokawa, H., Kokaji, S., "A motion planning method for a self-reconfigurable modular robot," in Proceedings of 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001), pp. 606-612, Hawaii, USA.

[3] Shen, W.-M., B. Salemi, and P. Will, Hormone-Inspired Adaptive Communication and Distributed Control for CONRO Self-Reconfigurable Robots, *IEEE Transactions on Robotics and Automation*, 18(5), October, 2002.

[4] Shen et al. "Distributed control and coordination of autonomous agents in a dynamic reconfigurable system," in US Patent 6,636,781 B1. 2003.

[5] Støy, K., W.-M. Shen, P. Will, "Implementing configuration dependent gaits in self-reconfigurable robots," International Conference on Robotics and Automation, Taiwan, 2003.

[6] Behnam Salemi, Peter Will, Wei-Min Shen, "Distributed task negotiation in self-reconfigurable robots," International Conference on Intelligent Robots and Systems. Las Vegas, October 2003.

[7] Arkin, R.C., R. R. Murphy, "Autonomous navigation in a manufacturing environment," IEEE Trans. on Robotics and Automation, 1990. 6(4): pp. 445-454.

[8] Nilsson, M. "Symmetric docking in 2D: A bound on self-alignable offsets," in Proc. IASTED'99: Robotics and Automation. 1999. Santa Barbara, California.

[9] Roufas, K., Y. Zhang, D. Duff, M. Yim. "Six degree of freedom sensing for docking using IR LED emitters and receivers," in The 7th Intl. Symp. on Experimental Robotics. 2000. Hawaii.

[10] Fukuda, T., and Y. Kawauchi, "Method of autonomous approach, docking and detaching between cells for dynamically reconfigurable robotic system CEBOT," Japan Soc. Mech. Eng. (JSME) Int'l Journal, 1990. 33(2): pp. 263-268.

[11] Murata, S., E. Yoshida, H. Kurokawa, K. Tomita, S. Kokaji, "Self-repairing mechanical systems," in Autonomous Robots, 2001. vol. 10, pp. 7-21.

[12] R. Fitch, Z. Butler, D. Rus, Reconfiguration Planning for Heterogeneous Self-reconfiguring Robots. International Conference on Intelligent Robots and Systems. Las Vegas, October 2003.

[13] Rus, D., M. Vona, Crystalline Robots: Self-Reconfiguration with Compressible Unit Modules. J. of Autonomous Robots, 2001. 10(1): p. 107-124.

[14] Rubenstein, M., Shen, W.-M., Will, P., "Docking among independent and autonomous CONRO self-reconfigurable robots," in International Conference on Robotics and Automata 2004. In press.

[15] B. Khoshnevis, P. Will, W.-M. Shen, "Highly compliant and self-tightening docking modules for precise and fast connection of self-reconfigurable robots," International Conference on Robotics and Automation, Taiwan, 2003.

[16] Stoy, K., W.-M. Shen, P. Will. Using role-based control to produce locomotion in chain-type self-reconfigurable robots, *IEEE Trans on Mechatronics*, 7(4), 410- 417.

[17] Shen, W.-M. and P. Will, "Docking in Self-Reconfigurable Robots", *IROS 2001*.