Sensor Computations in Modular Self Reconfigurable Robots

Ying Zhang, Kimon Roufas, Craig Eldershaw, Mark Yim, Dave Duff
Palo Alto Research Center
Palo Alto, CA 94304
{yzhang, kroufas, celdersh, yim, dduff}@parc.com

Abstract: Sensors play important roles in automatic systems; smart systems have to be equipped with smart sensors. PolyBot, a modular self-reconfigurable robot developed at the Palo Alto Research Center, is designed with a rich set of sensors in each of its 5cm-cubed modules. These include infrared (IR), accelerometers, potentiometers, force, and touch sensors. These sensors are used: to determine the current state of the system and its environment; to obtain the six degree-of-freedom offset between two docking plates for automatic reconfiguration; to select the right gait for locomotion; and to trigger different behavior modes in response to different terrain conditions. Sensor computations are computational methods that, given raw sensor data, extract or deduce the state information about the system. In general, there are two types of sensor computation, forward computation and inverse computation. Analogous to forward and inverse kinematics, forward sensor computation obtains state information directly from sensor data; inverse sensor computation obtains state information by solving a constraint or optimization problem using either closed-form or numerical methods. This paper focuses on two interesting sensor computations in PolyBot: IR 6 DOF ranging and accelerometer 2 DOF orientation.

1. Introduction

A robot's ability to sense its world and its own state, so as to adapt to the changing environments and to compensate for its imperfect internal model, is what makes a robot "smart". Sensors play an important role in making an automatic system a smart system [1]. PolyBot [2], a modular self-reconfigurable robot developed at the Palo Alto Research Center (PARC), will consist of 100+ connected 5cm-cubed modules, each of which is equipped with a rich set of sensors, actuators and a micro-controller. Figure 1 shows a PolyBot module and its major sensors and actuators.

Sensor computations are computational methods that, given raw sensor data, extract and deduce state information about the system. In general, there are two types of sensor computation: forward sensor computation and inverse sensor computation. Analogous to forward and inverse kinematics, forward sensor computation obtains state information directly from sensor data; inverse sensor computation obtains state information by solving a constraint or optimization problem using either closed-form or numerical methods.

Two interesting sensors in PolyBot are the IR ranging sensors and the accelerometer orientation sensors. For self-reconfiguration, when two modules choose to connect to each other (dock), they require accurate knowledge of their relative positions and orientations. IR ranging sensors are used to obtain the six degree-of-freedom (DOF) offset information between two docking plates. Accelerometer sensors are used to compute module orientations relative to gravity. Such information can be used for configuration management, locomotion gait selection, and in specialized gaits. This paper focuses on these two types of sensors, discussing their computational methods and novel applications.

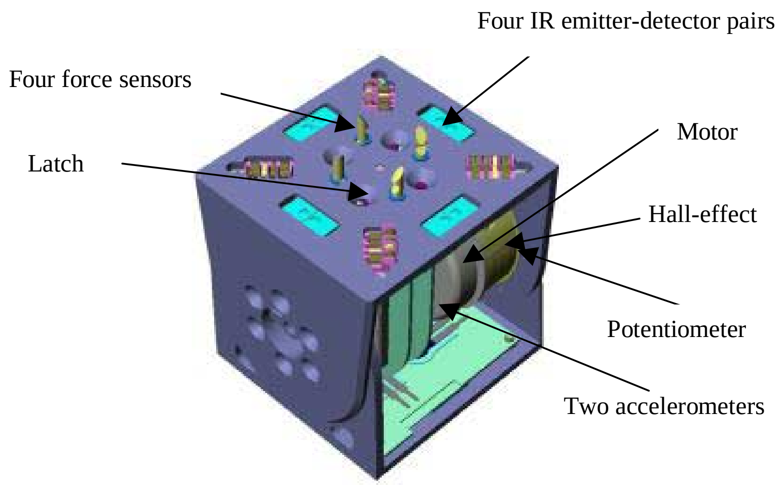

Figure 1. Sensors and actuators on a PolyBot module

This paper is organized as follows: Section 2 presents the IR 6 DOF ranging mechanism, with an inverse computation for obtaining 6 DOF offset information and an alternative method using forward computation for obtaining 6 DOF directional offset information. Section 3 illustrates the computational method for obtaining a 2 DOF orientation vector for any module, and its applications to gait selection and specialized gait generation. Section 4 concludes the paper.

2. IR 6 DOF Ranging

The 6 DOF offset sensing between two plates is critical for automatic active docking of self-reconfigurable robot systems such as PolyBot. Automated active docking requires that the positional offset errors are measured and then corrected by an automated control system.

The PolyBot system uses many modules, all with identical docking mechanisms, or *interface plates*. Since two interface plates that may dock with each other are identical, they need to have hermaphroditic connection mechanisms. Other systems such as [3][4] may have male and female connection mechanisms, however, the sensing method described in this paper is general and extends to those systems as well.

There has been an implementation of 6 DOF sensing system, using four commercial-off-the-shelf (COTS) infrared (IR) light emitting diodes (LED) and two COTS IR detectors on each of the opposing plates [5]. This paper describes a new improved implementation. Two computational methods will be discussed in detail: one is inverse computation using the least squares minimization, which obtains 6 DOF offset, the other is forward computation using "information balancing" based purely on the geometric symmetry of the emitters and detectors, which obtains 6 DOF directional offset. Experiments have been conducted indicating that such a setup is able to give 6 DOF offset (x, y, z, pitch, roll, and yaw) within a range. The computational methods are general and apply to other configurations of emitter and detector parts and configurations.

2.1 Mechanical and Electronic Design

In the previous mechanical design of the system [5], four LEDs are placed on the four corners and two detectors are placed along the centerline on an interface plate. The new design places four emitter-detector pairs on the center of four edges. Figure 2 shows the new mechanical design of the plate, the dot denotes an emitter and the circle denotes a detector. The new design has the property that when two centered faces are closer, the intensities received from the corresponding emitters are larger; while in the previous design, the intensities are all diminished and eventually vanished due to large emitter-detector angles. The new design also enables local communication for two connected modules. This allows recognition of which of the two plates are connected and which of the four possible orientations the two connected plates are in, an important step for automatic configuration recognition in the initialization of a modular self-reconfigurable system.

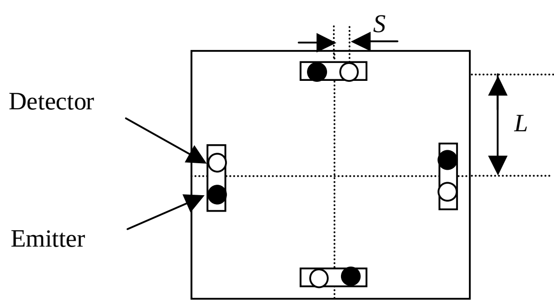

Figure 2. Mechanical design of the IR 6 DOF sensing device on a PolyBot faceplate

In the previous electronic design, each emitter from both plates is lit in sequence and data are taken from both plates to calculate the offset information; data from one plate is not sufficient for obtaining the offset. In the new electronic design, each emitter from one plate is lit in sequence and each detector in the opposite plate gets the intensity samples from each emitter at a distinct time. Data samples from one plate are sufficient to obtain the offset. A reading of the ambient light is taken at each transition when all the emitters are briefly off. At the end of a time period, each of the detectors will have four readings from their opposing emitters and four ambient readings, totaling 32 measurements from one plate. The ambient IR readings are subtracted from the immediately preceding sample to make the system more robust to rapid changes in ambient light conditions. Hence it ends up with 16 pieces of data, from which, the 6 DOF offset is computed.

2.2 IR Intensity Model

The computational method for obtaining 6 DOF offset utilizes the fact that the intensity detected by a detector is a function of the distance and/or angles between the emitter and detector, i.e., $I = f(e, r, d)$ where I is the intensity reading, e and r are emitter and detector angles, respectively, and d is the distance between the emitter and the detector. An accurate model can be obtained by model fitting for a given emitter and detector pair. In the previous implementation, the emitter-detector pair is mostly sensitive to angles rather than distances. The new implementation uses an emitter-detector pair that is less sensitive to emitter and detector angles, but more sensitive to distance. For better results, the emitter intensity and the detector sensitivity have been tuned so that the distance curve has a wide sensitivity range. Using the model $f(d) = A/d^2+O$, O is measured to be 35, A is obtained via a least square curve fitting, which is 217. Figure 3 shows both the measured value and those calculated from the model, of the IR emitter-detector pair used for the new design.

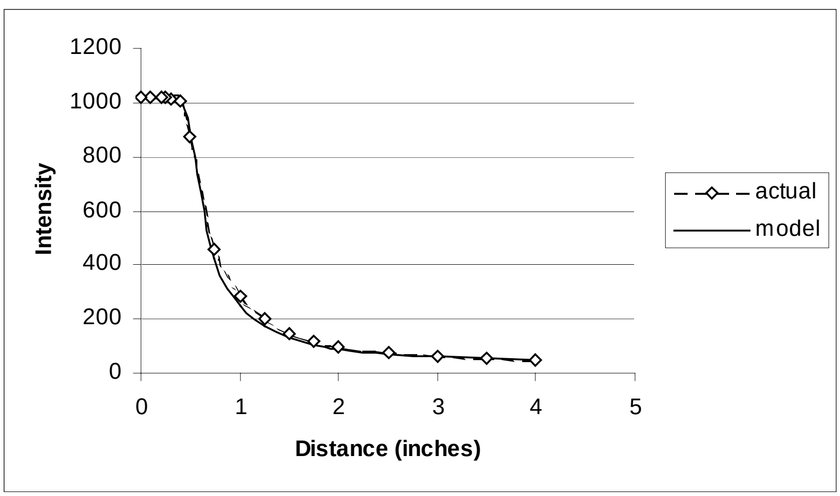

Figure 3: The actual and model intensity curves

2.3 6 DOF Ranging Computations

Each interface plate has a local coordinate frame, as shown in Figure 3 (in this case Plate 1 and Plate 2 are facing each other). Given an offset between the two plates, the spatial relationship between every pair of emitter and detector is determined.

Let S be the shorter distance from the emitter or detector to the center of the plate, and L be the longer distance to the center of the plate as in Figure 2. The coordinates of the four detectors are <0, *L*, -*S*>, <0, *S*, *L*>, <0, -*L*, *S*> and <0, -*S*, -*L*>. The coordinates of the four emitters are <0, -*L*, -*S*>, <0, -*S*, *L*>, <0, *L*, *S*> and <0, *S*, -*L*>. Let <*x*, *y*, z, α, β, γ> be the offset of the frame of plate 2 with respect to the frame of plate 1 (e.g., in the case of two plates facing each other, the offset is <*x*, *0*,

$0, \pi, 0, 0$>) and let T be the 4x4 transformation matrix from plate 1 to plate 2 obtained by the offset, and R be the 3x3 rotation matrix of T. The norm of plate 1 is <1, 0, 0> and the norm of the plate 2 in plate 1 coordinates is R<1, 0, 0>. Let $o=\langle x_d, y_d, z_d\rangle$ be the coordinate of the detector in its own frame in plate 1 and $p=\langle x_e, y_e, z_e\rangle$ be the coordinate of the emitter in its own frame in plate 2, the position of the emitter in the frame of plate 1 is $q = Tp$.

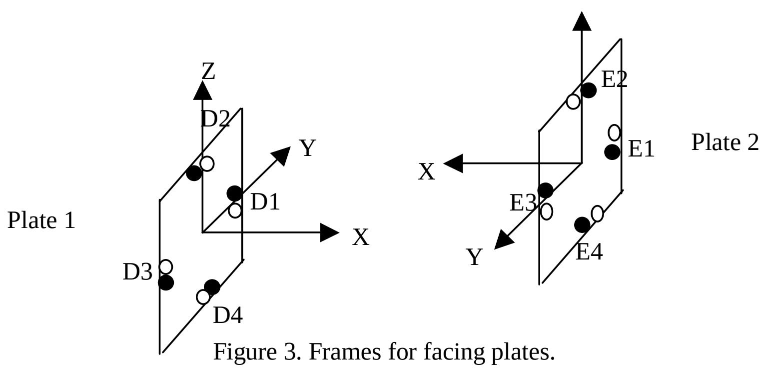

Figure 3. Frames for facing plates.

Given two points in space, o and q, and the norms of their plates, n_o and n_q, the distance between them is $|q\text{-}o|$, the angle at o is $\cos^{-1}(\frac{n_o \cdot (q-o)}{|q-o|})$ and the angle at q is $\cos^{-1}(\frac{n_q \cdot (o-q)}{|q-o|})$. Therefore, the emitter and detector angles as well as the distance between the detector and the emitter can be obtained for each of the sixteen pairs of emitters and detectors. Given the IR intensity model obtained in the previous section, a model is obtained from each 6 DOF offset between two plates to 16 readings of intensities, i.e., $I_i = f_i(x, y, z, \alpha, \beta, \gamma)$ for i=1 to 16.

2.3.1 6 DOF position sensing: inverse computation

Theoretically, the problem of 6 DOF offset estimation is an inverse computation problem, i.e., solving $\langle x, y, z, \alpha, \beta, \gamma\rangle$ given sixteen data readings, based on IR intensity model and emitter-detector positions. A least square minimization can be used for fitting the data. In particular, let I_i, i =1..16, be the sixteen readings and let E be an energy function to be minimized, $E = \frac{1}{2}\sum_1^{16}(I_i - f_i(x,y,z,\alpha,\beta,\gamma))^2$ which transforms to six equations: $\frac{\partial E}{\partial p} = \sum_1^{16}(I_i - f_i(x,y,z,\alpha,\beta,\gamma))\frac{\partial f_i}{\partial p} = 0$ where p is x, y, z, α, β, or γ.

This set of equations can then be solved using Newton's method. Here, Singular Value Decomposition (SVD) is used for solving linear equations at each Newton step. The use of SVD greatly reduces the risk of reaching a singularity (a common problem when inverting matrices). It also achieves a better result in both under- (minimum change) and over-constrained (minimum error) situations.

This method did not work well for the previous implementation [5]; but does work quite well for the new implementation. The key to make this method work is the IR model. In the previous design, the particular emitter and detector pairs used were not ideal. Firstly, it is only sensitive at very close ranges, and secondly, the slope in the valid range is too steep, which makes the sensor data overly sensitive. The new pair is less sensitive to angles, but more sensitive to distance; and also the emitter-detector pair has been tuned so that valid readings can be taken over a wider range of distances. Experiments show that the current setting works quite well from distance 0.3 inches to 1.5 inches. Beyond the range, directional 6 DOF offset sensing will be used.

2.3.2 Directional 6 DOF offset sensing: forward computation

In the closed loop control of the docking process, it is not necessary to have accurate 6 DOF position sensing all the time, as long as (1) it can tell the direction of the offset and (2) it can tell if the offset between the plates are small enough; (1) is used to guide the motion of the plates and (2) is to trigger the latches in the plates to open and close at approaching times during the docking procedure.

A centering method has been developed based on the idea of the signal "balancing" when the plates are centered and facing each other. Let *Vij* represent a reading where *i* is the emitter identifier (1-4) and j is the detector identifier (1-4), see Figure 3. When two plates are centered and facing each other, a set of equations hold, e.g., *V11=V33*, *V22=V44*, etc. In practice, even when the two plates are exactly centered, the equations may not hold because of noise and slight variations when mechanically assembling the plates. The difference, however, can be used as a guideline for the directional offset information. For example, *(V32-V12)+(V34-V14)+(V21-V23)+(V41-V43)* gives offset in Y direction, while *(V43-V23)+(V41-V21)+(V32-V34)+(V12-V14)* gives directional offset in Z direction. This method has been used to successfully dock two plates in a plane, i.e., a special case with 3D offset, even in the previous design [5].

To follow this path further, five "balancing" equations are discovered, each of which corresponds to an invariant with respect to all but one of 6 DOF offset:

1. **Y Variant** ($x, z, \alpha, \beta, \gamma$ invariant): *V32 + V21 + V34 + V41= V23 + V12 + V43 + V14*.
2. **Z Variant** ($x, y, \alpha, \beta, \gamma$ invariant): *V43 + V32 + V41 + V12 = V23 + V34 + V21 + V14*.
3. $\boldsymbol{\alpha}$ **Variant** (x, y, z, β, γ invariant): *V11=V33*.
4. $\boldsymbol{\beta}$ **Variant** (x, y, z, α, γ *invariant*): *V22=V44*.
5. $\boldsymbol{\gamma}$ **Variant** (x, y, z, α, β invariant): *V41 + V12 + V23 + V34 = V21 + V32 + V43 + V14*.

A minimization method is developed that can be used for any equation. For example, for equation *A*=*B*, an energy function $E = \frac{1}{2}(A - B)^2$ is derived. Note that this energy function does not contain the explicit IR model as the one used for the 6 DOF offset position sensing in the inverse computation. The goal of centering is to move to the direction where the energy function can be minimized. In order to minimize *E*, calculate $\frac{\partial E}{\partial p} = (A - B)(\frac{\partial A}{\partial p} - \frac{\partial B}{\partial p})$, $\frac{\partial^2 E}{\partial^2 p} \approx (\frac{\partial A}{\partial p} - \frac{\partial B}{\partial p})^2$, in which, p is y, z, α, β or γ. Setting $\Delta p = -\frac{\partial E}{\partial p} / \frac{\partial^2 E}{\partial^2 p}$ gives the direction of the offset movement required to minimize the energy function obtained from the equation. Because each equation corresponds to exactly one variant, it is used to calculate the offset of that variant, i.e., equation 1 to calculate y, equation 2 to calculate z, equation 3 to calculate α, equation 4 to calculate β and equation 5 to calculate γ. To calculate x, use energy function $E = -\frac{1}{2} \sum_{i=1,2,3,4} V_{ii}^2$, since all the four readings go to maximum when x approaches 0 in centered position. For simplicity, assuming the plates are centered, then $\Delta x = -\frac{\partial E}{\partial x} / \frac{\partial^2 E}{\partial^2 x}$, where $\frac{\partial E}{\partial x} = -\sum_{i=1,2,3,4} (Vii \frac{\partial Vii}{\partial x})$ and $\frac{\partial^2 E}{\partial^2 x} \approx -\sum_{i=1,2,3,4} (\frac{\partial Vii}{\partial x})^2$.

In the case where the IR curve is flat (distance close to 0 or larger than 5 inches), direct estimation is used, i.e., $y = V32 + V21 + V34 + V41 - V23 + V12 + V43 + V14$, $z = V43 + V32 + V41 + V12 - V23 + V34 + V21 + V14$, $\alpha = \mathrm{V11} - \mathrm{V33}$, $\beta = V22 - V44$, $\gamma = \mathrm{V21 + V32 + V43 + V14 - V41 + V12 + V23 + V34}$. The threshold to decide the direction depends on the noise level. The estimation of x can be obtained by calculating $V = (V11+V22+V33+V44)/4$ and finding the corresponding distance value from the IR model's curve.

As this method does not involve solving equations, rather, results are derived directly from raw sensor data; it is a forward computation method. This method is used together with the inverse computation method to make a robust estimation of 6 DOF offset.

3. Accelerometer 2 DOF Orientation

Accelerometers can be used for obtaining orientations of modules with respect to gravity. Moreover, it can be used for configuration management, gait selection and specialized gait generation for modular self-reconfigurable robots.

3.1 Mechanical and Electronic Design

There are two accelerometers [6] on each PolyBot module. Figure 4 shows the coordinate frame of a module, where Z is aligned with the rotation axis. The two accelerometers are mounted on two orthogonal planes, one on X-Y plane, one on Y-Z plane (Figure 4). The X-Y plane is fixed to the module base; the Y-Z plane can be rotated about the Z-axis. There are four readings, two from each accelerometer. The value of an accelerometer reading reflects the force/acceleration along the axis.

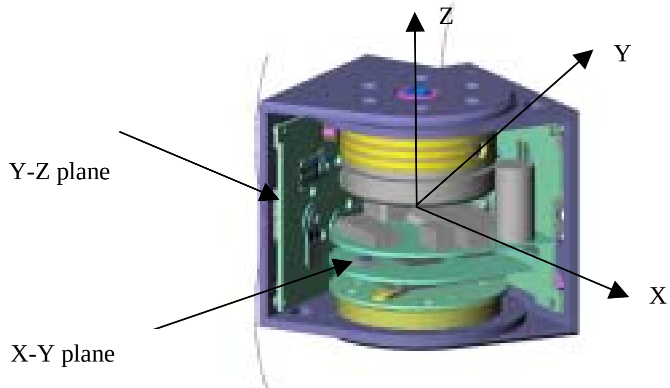

Figure 4: Two accelerometers, one mounted on X-Y plane, one on Y-Z plane

3.2 2 DOF Orientation Computation

Given four accelerometer readings, the β and γ of a module's orientation can be calculated. The calculation is an inverse computation. However, in this case, the closed form solution is obtained.

Assume that the module orientation is $<\alpha, \beta, \gamma>$ and the module (Y-Z plane) rotate about the Z-axis by θ degrees. The rotational transformation matrix is:

$$\begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix} \tag{1}$$

Let d_0, d_1, d_2, d_3 be the accelerometer readings along X-, Y-, Y^{θ}- and Z^{θ}-axis, respectively. If the module is static, i.e., only gravity is measured, the readings should be: $d_0 = -s\beta, d_1 = c\beta s\gamma, d_2 = s\beta s\theta + c\beta s\gamma c\theta, d_3 = c\beta c\gamma$. This yields four equations and three variables. If, however, the module is not static, there are four equations with more than three variables, possibly including α.. Assume modules are static (no motion), a closed-form solution for β, γ and θ can be calculated as follows: $\beta = \sin^{-1}(-d_0), \gamma = \tan^{-1}(d_1, d_3)$ when $c\beta$ is not zero. If $c\beta$ is zero, γ cannot be calculated. To calculate θ, solve equation $d_2 = -d_0 s\theta + d_1 c\theta$ for θ. If $d_0^2 + d_1^2 < d_2^2$ or $d_0^2 + d_1^2 = 0$, θ cannot be calculated, and so it is only a redundant constraint for robustness, the actual θ is obtained via joint sensors. In many applications, as discussed in next section, the raw readings can be used directly, instead of calculating the 2 DOF orientations.

3.3 Novel Applications to 2 DOF Orientation

Accelerometer readings are important to make modular robot self-aware about its current state in the environment, e.g., a loop track falls down (Figure 5), a spider is upside down or a snake is lying sideways (Figure 6). Such diagnostic information can be used in various ways. In some cases, a reconfiguration or recovery sequence can be triggered, in some other cases, a simple change of gaits or parameters is sufficient.

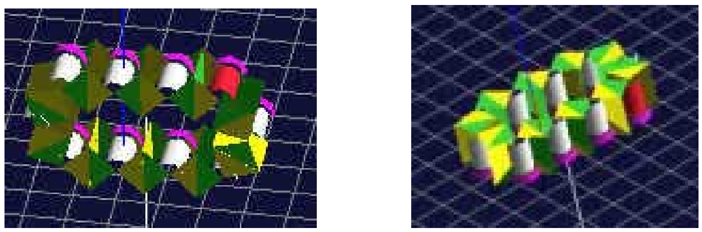

Figure 5: Loop and falling loop

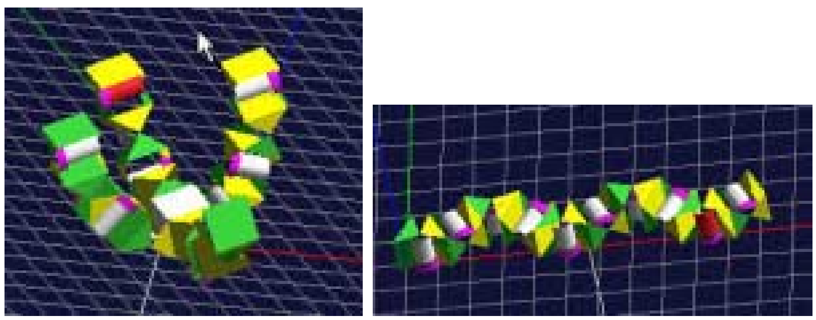

Figure 6. Spider upside down and snake lying on sideways

In most situations, the magnitude of the projection of the rotation axis (Z-axis) on the gravity vector should be obtained. Using the rotational transformation matrix (1), it is $c\beta c\gamma$, which is d_3. To determine if a loop is falling down or not, it is suffice to test if all d_3's are small (close to zero); if they are, the loop is standing; otherwise the loop has fallen. Theoretically, d_3 from one module is sufficient, however, using extra readings will increase robustness. Similarly, to see if a spider is upside down can be done by checking if the d_3 reading from the center node has changed its sign. In the case of a fallen loop, a reconfiguration may be required. In the case of an upside down spider, changes of signs of some joint angles would fix the problem. For the situation of a snake configuration, if modules are connected orthogonally as shown in Figure 6, d_3 can be used for selecting gaits for the modules, i.e., if d_3 of a module is small, it should run a sinusoid gait, otherwise, it should run a bend gait, namely first bend and then unbend, when turning is issued. In the case of a sinusoid gait, the sign of $cos\beta sin\gamma$, i.e. the projection of the local Y-axis to reference Y axis, can be used to set the sign of the joint angle, so that if it is issued a positive value, the module bends up (counter-clockwise). In the case of a bend gait, the sign of $cos\alpha cos\gamma$, i.e. the projection of the local Y-axis to reference Z-axis, can be used to set the sign of the joint angle, so that if it is issued a positive value, the module bends left (counter-clockwise). Here α is either 0 or π, depending on the plate connection which can be obtained by local communication using IR.

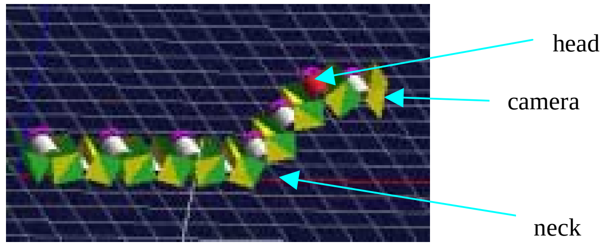

Figure 7: A snake holding its head up

Accelerometer sensors can also be used for specialized gaits, such as holding the head in a straight level pose for camera, while the rest of the snake body running snake gaits (Figure 7). To do that, two specialized modules are configured, one “neck”, one “head”. Both modules run a closed-loop control to regulate the desired orientation of its plate, i.e., the projection of X-axis to the gravity vector, for example, 0.8 for neck, 0 for head, shown in Figure 7. Given d_0, d_1 from the accelerometer reading, the projection of X-axis of its plate to the reference Z-axis (gravity) can be calculated as $f(\theta)=d_0s\theta+d_1c\theta$. Given d as desired projection, using energy based strategy, control is to minimize $(d-f(\theta))^2$, the control law using the gradient method, $d\theta=(d-f(\theta))*f'(\theta)$, i.e., $d\theta=(d-d_0s\theta-d_1c\theta)*(-d_0c\theta-d_1s\theta)$, is derived, where θ can be obtained from the joint sensor.

4. Conclusions

This paper has presented two sensor computations used in PolyBot, a modular self-reconfigurable robot. These are IR 6 DOF ranging and accelerometer 2 DOF orientation. For the IR ranging, an inverse computation is applied so that the 6 DOF offset position is obtained from 16 measurements. The new mechanical and electronic design has improved the performance of the computational method, through using a new geometric configuration and selecting and tuning IR emitter-detector pairs. The directional offset sensing is also more effective than in the previous design, since invariant equations are discovered for each offset individually. For the accelerometer 2 DOF orientation, a method has been implemented for obtaining the 2 DOF orientation vector of a module and applying this: to detect various conditions of the robot, to select gaits and self-recovery sequence and to generate closed-loop control law.

From computational point of view, IR 6 DOF ranging is a typical inverse computation, however, as an alternative, forward computation using the idea of signal balancing based on the geometric symmetry of the emitter-detector configuration can be used for obtaining directional offset information quickly and robustly. Accelerometer 2 DOF computation is an inverse computation in nature; however, a closed-form solution can be obtained for static situations, where only gravity is detected. Both sensors play important roles in modular self-reconfigurable robots.

Acknowledgements:

This work is funded in part by the Defense Advanced Research Project Agency (DARPA) contract # MDA972-98-C-0009.

References

[1] Everett, H. R., "Sensors for Mobile Robots: Theory and Applications", A K Peters, Ltd. 1995.

[2] Yim, M., Duff, D.G., Roufas, K., "PolyBot: a Modular Reconfigurable Robot", in *Proc. of the 2000 IEEE Intl. Conf. on Robotics and Automation*, 2000.

[3] Will, P., Castano, A. and Shen W-M., "Robot modularity for self-reconfiguration," in *Proc.of SPIE Sensor Fusion and Decentralized Control in Robotic Systems II*, Vol. 3839, pp. 236-245, 1999.

[4] Fukuda, T., Nakagawa, S., Kawauchi, Y., and Buss, M., "Structure decision method for self organising robots based on cell structures-CEBOT'", in *Proc. of the 1989 IEEE Intl. Conf. on Robotics and Automation*, 1989.

[5] Roufas, K., Zhang, Y., Duff, D., and Yim, M., "Six Degree of Freedom of Sensing for Docking using IR LED Emitters and Detectors", Experimental Robotics VII, Lecture Notes in Control and Information Sciences 271, Daniela Rus and Sanjiv Singh Eds. Springer, 2001. Pages: 91-100

[6] Accelerometer Data Sheet, http://www.analog.com/productSelection/pdf/ADXL202E.pdf