Helical Controller for Modular Snake Robot with Non-holonomic Constraint

Pongsakorn Polchankajorn and Thavida Maneewarn
Field Institute of roBOtics
King Mongkut's University of Technology Thonburi
Bangkok, Thailand
dome@fibo.kmutt.ac.th, praew@fibo.kmutt.ac.th

***Abstract* - In order to control a modular wheeled snake robot, there is a complexity of generating a control solution that has to satisfies non-holonomic constraints which depends on the robot configuration. For a pole climbing application, in which the modular snake robot is required to climb a cylindrical pole vertically, we proposed a control method for the modular snake. By generating the control input such that the pose of the robot follows a helical curve configuration, the control input is constrained to be within the null space of the non-holonomic constraint where the desired wheel velocity can be mapped directly to one-dimensional control input signal of the robot. The proposed controller was successfully implemented and tested in our modular pole climbing snake robot.**

Index Terms - Modular Snake Robot, Non-holonomic Control, Pole Climbing.

I. INTRODUCTION

A modular snake robot is the robot that is constructed in the modular structure where all links and joints have the same design. This type of robot is also considered hyper-redundant [1] because the robot degrees of freedom are much larger when compare to the number of dimensions of the task. Most research in this area concern with the generation of snake-like or crawling motion [2, 4]. Some research focus on the inverse kinematic problem which becomes very complex due to the hyper-redundant nature of the robot. By adding the wheel to each module of the snake robot, additional complexity arises from the non-holonomic constraints created from the wheel.

Our specific interest in a modular snake robot is in pole climbing applications. There are needs in service and construction industry for a robot that can climb a pole or a pipe or even a tree. Thus, we have designed and constructed a modular wheeled snake robot that can be used for pole climbing application because of its lightweight and its ability to adapt to various size of structure. There has been many works in climbing gait of snake robot prior to our study. The first grasping gait that applied from travelling wave amplitude constant (TWAC) was presented by Chirikjian and Burdick [1] they use this gait to control the snake robot to coil around the pole. After that, numerical solution for discretized 3-D path and inverse kinematics for universal joint snake robot was presented by Anderson [3]. Then Choset [4, 5] applied piecewise differentiable gait to create climbing gait of the snake robot. This gait can climb by rolling up vertically with helical configuration. He also suggested the Toroidal Skin Drive (TSD) which the robot's skin rolling laterally instead of using wheels. He had the locomotion experiment of this driving type such as crossing the gap, climbing the pole in helical trajectory, etc. Seirei Industry [5] built the commercial pruning machine to cut tree's branches. This machine also used the helical rolling concept to climb the tree using multiple wheels that lock around the tree's trunk and roll up the tree.

Our research is focus on the modular wheeled snake robot. With the wheeled modular snake robot, the load from vertical climbing can be reducing partially from the non-holonomic constraints created by the wheel. In this paper, a helical path controller is proposed to use for creating the simple control algorithm for the robot with complex non-holonomic constraints. In the next sections, we will discuss about system's overview e.g. the structure of robot and control system, dynamics model of the robot, robot's motion constraint, the proposed helical path controller, and the results from our experiment.

II. SYSTEM'S OVERVIEW

A. *Robot's structure*

The robot's structure of modular snake robot consists of 7 links, 6 joints. Each joint has 3 degrees of freedom. Links of the robot has 19 cm long, 11 cm wide and 7.5 cm height. Each link attached with two active wheels. Wheel sizes are 10 cm diameter, 1 cm thickness. The robot structure is all made from aluminium. Each joint use three commercial servo motors (Robotis DX-117) and use one motor (Robotis RX-28) to drive link's wheels. The total weight of the robot is approximately 3 kg.

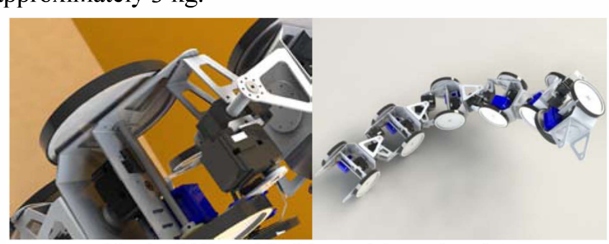

Fig. 1. CAD model of the modular snake robot.

B. *Control system design*

The motor that use in the robot is digital servo motor, which has low-level position control, we only send the position protocol via RS-485 to control the motor. The position of each motor is calculated and sent from the computer as high-level control. The microcontroller (ATMEL ATMEGA128) is used to convert the position data that received from computer via RS-232 to motor's protocol packet and send to the motor. Robot's power is sent by external power supply. The simple control system diagram is shown in fig. 2

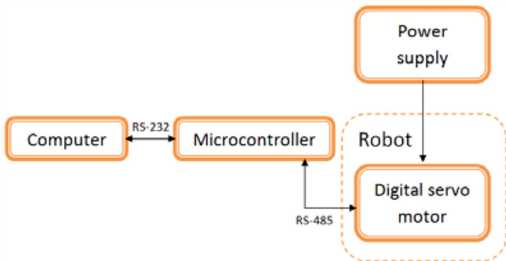

Fig. 2. System design of the modular snake robot.

C. *Reference frame description*

The world's reference frame was set so that the axis y is normal to the ground plane. Each link's reference frame can be described as follow. X axis located along the link's length. So the angle that rotates around x axis is the roll angle. Pitch angle and yaw angle are the angles that rotate around z axis and y axis respectively.

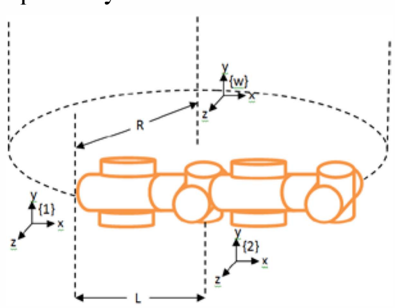

Fig. 3. Robot's frame assignment.

III. DYNAMIC MODEL OF THE ROBOT

The dynamic model of the snake robot is derived from Lagrangian method. Robot's structure is assumed to be a rigid. The robot's wheels are also assumed to be in a non-slip condition. The link's dynamic model of the robot can be written in general form like equation 1.

$$\tau_i = M_i(q_i)\ddot{q}_i + V_i(q_i,\dot{q}_i) + G_i(q_i) + J_i(q_i)\lambda \qquad (1)$$

With $q_i = [\alpha_i \quad \beta_i \quad \gamma_i \quad \theta_i]^T$ and $\tau_i = [\tau_{\alpha_i} \quad \tau_{\beta_i} \quad \tau_{\gamma_i} \quad \tau_{\theta_i}]^T$. Assume i as the number of link.

The angles α_i, β_i, γ_i, θ_i, τ_{α_i}, τ_{β_i}, τ_{γ_i}, and τ_{θ_i} are the roll angle, pitch angle, yaw angle, rolling angle of robot's wheel, roll torque, pitch torque, yaw torque, and wheel's torque, respectively. $J_i(q_i)\lambda$ is the non-holonomic constraint term that has the null space of linear-velocity transform

$$J_i^T(q_i)\dot{q} = 0 \qquad (2)$$

Let,

I_x, I_y, I_z : Moment of inertia of the whole link,
I_w : Moment of inertia of the wheels,
l : Link's length,
m : Link's mass,
r : Wheel's radius, and
g : Acceleration due to the gravity

Then the matrices used in the link's model (equation 1) are,

$$M_i = \begin{bmatrix} M_{11} & 0 & 0 & 0 \\ 0 & M_{22} & 0 & 0 \\ 0 & 0 & M_{33} & 0 \\ 0 & 0 & 0 & M_{44} \end{bmatrix} \text{with}$$

$M_{11} = ml^2 + I_x$, $M_{33} = ml^2 + I_y$,

$M_{22} = ml^2 + I_z$, $M_{44} = mlr^2 + I_w$,

$V_i = 0$, and

$G_i = [G_1 \quad G_2 \quad 0 \quad 0]^T$ with

$G_1 = -m\lg\sin\alpha_i\cos\beta_i$,

$G_2 = -m\lg\cos\alpha_i\sin\beta_i$.

After that, global dynamic model of the robot can be express in matrix form as follow.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_n \end{bmatrix}_{4n\times1} = \begin{bmatrix} M_1 & 0 & 0 & 0 \\ 0 & M_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & M_n \end{bmatrix}_{4n\times4n} \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_n \end{bmatrix}_{4n\times1} + \begin{bmatrix} G_1 \\ G_2 \\ \vdots \\ G_n \end{bmatrix}_{4n\times1} \qquad (3)$$

From the local matrix, one link of the robot has four control parameters and joint variables. So, the robot that consists of n links totally has $4n$ control parameters and joint variables. In the task space that of the robot, we will transform the joint coordinate to Cartesian coordinate as follow.

$$\tau_i = M_i(X_i)\ddot{X}_i + V_i(X_i,\dot{X}_i) + G_i(X_i) + J_i(X_i)\lambda \qquad (4)$$

With $X_i = [x_i \quad y_i \quad z_i \quad \theta_i]^T$

IV. ROBOT'S MOTION CONSTRAINT

From the structure of the robot that uses active wheels in each links, the assumption about the constraint of the wheels that prevent the links from side slipping is considered. We can be derived the constraint of the robot in two types: holonomic and non-holonomic constraint.

Holonomic constraint can be write in three equations

$$x_i = x + \sum_{j=1}^{i-1} LC\beta_i C\gamma_i + \frac{L}{2}C\beta_i C\gamma_i + \theta r C\beta_i C\gamma_i \qquad (5)$$

$$y_i = y + \sum_{j=1}^{i-1} L(C\alpha_i S\beta_i C\gamma_i + S\alpha_i S\gamma_i) + \frac{L}{2}(C\alpha_i S\beta_i C\gamma_i + S\alpha_i S\gamma_i) \qquad (6)$$

$$z_i = z + \sum_{j=1}^{i-1} L(S\alpha_i S\beta_i C\gamma_i + C\alpha_i S\gamma_i) + \frac{L}{2}(S\alpha_i S\beta_i C\gamma_i + C\alpha_i S\gamma_i) \qquad (7)$$

We will get the trajectory of the robot by differentiate the equation 5-7.

$$\dot{x}_i = \dot{x} - \sum_{j=1}^{i-1} L\left(\dot{\gamma} C\beta_i S\gamma_i + \dot{\beta} S\beta_i C\gamma_i\right) - \frac{L}{2}\left(\dot{\gamma} C\beta_i S\gamma_i + \dot{\beta} S\beta_i C\gamma_i\right) + r\left(\dot{\theta} C\beta_i C\gamma_i - \theta_i \dot{\gamma} C\beta_i S\gamma_i - \theta_i \dot{\beta} S\beta_i C\gamma_i\right) \quad (8)$$

$$\dot{y}_i = \dot{y} + \sum_{j=1}^{i-1} L\begin{pmatrix} \dot{\beta}_i C\alpha_i C\beta_i C\gamma_i - \dot{\gamma}_i C\alpha_i S\beta_i S\gamma_i \\ -\dot{\alpha}_i S\alpha_i S\beta_i C\gamma_i + \dot{\alpha}_i C\alpha_i S\gamma_i + \dot{\gamma}_i S\alpha_i C\gamma_i \end{pmatrix} + \frac{L}{2}\begin{pmatrix} \dot{\beta}_i C\alpha_i C\beta_i C\gamma_i - \dot{\gamma}_i C\alpha_i S\beta_i S\gamma_i \\ -\dot{\alpha}_i S\alpha_i S\beta_i C\gamma_i + \dot{\alpha}_i C\alpha_i S\gamma_i + \dot{\gamma}_i S\alpha_i C\gamma_i \end{pmatrix} \quad (9)$$

$$\dot{z}_i = \dot{z} + \sum_{j=1}^{i-1} L\begin{pmatrix} -\dot{\gamma}_i S\alpha_i S\beta_i S\gamma_i + \dot{\alpha}_i C\alpha_i S\beta_i C\gamma_i \\ +\dot{\beta}_i S\alpha_i C\beta_i C\gamma_i + \dot{\alpha}_i S\alpha_i S\gamma_i - \dot{\gamma}_i C\alpha_i C\gamma_i \end{pmatrix} + \frac{L}{2}\begin{pmatrix} -\dot{\gamma}_i S\alpha_i S\beta_i S\gamma_i + \dot{\alpha}_i C\alpha_i S\beta_i C\gamma_i \\ +\dot{\beta}_i S\alpha_i C\beta_i C\gamma_i + \dot{\alpha}_i S\alpha_i S\gamma_i - \dot{\gamma}_i C\alpha_i C\gamma_i \end{pmatrix}$$

For non-holonomic constraint of each links, it simply describe in local coordinate of the link as

$$\dot{y}_{local} = 0 \quad (10)$$

In global coordinate consideration, we multiply with rotational term to transform local non-holonomic constraint as follow.

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \end{bmatrix} = \begin{bmatrix} {}^{W}_{i}R \end{bmatrix} \begin{bmatrix} 0 \\ \dot{y}_{local} \\ 0 \end{bmatrix} \quad (11)$$

V. Helical Path Controller

The objective of this research is to develop a modular wheeled snake robot that can be used for pole climbing application. For climbing a pole vertically, the snake robot must be able to solve two objectives. First, to be able to grasp the pole without falling vertically by gravitational force. Second, it should use the non-holonomic constraint to support the robot weight while grasping and climbing the pole. For the first objective, the robot must coil around the pole in order to generate the necessary amount of grasping force. For the second objective, moving path of the robot should have sufficient angle of elevation with the world's ground plane to move robot upward. Another consideration is the non-holonomic constraint in each link of the robot. Constraint of each link should not conflict with others.

We can use 3 equations to generate 3-D helix moving path as follow.

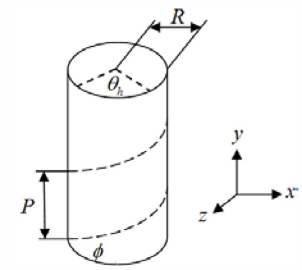

Fig. 4. Circular helix path.

$$z = R\cos(\theta) \quad (12)$$

$$x = R\sin(\theta) \quad (13)$$

$$y = \frac{P.\theta}{360} = \theta R \tan\phi \quad (14)$$

Where P, R, θ_h, and ϕ are the pitch distance, the path's radius on xz plane (top view), the angle parameter, and the helical pitch angle.

The helical path has two inputs: helical pitch angle and helical radius. After we define helical path input, we can discretize the helical curve to generate the orientation of the link's frame respect to the world's frame in rotation matrix form. Because the helical path is generated from the periodic function, we calculate the orientation of the frame with by the law of tangent between helical radius, pitch angle, and link length. We then use the relation of the rotation matrix to calculate the rotation matrix of links i with respect to the frame of the link i – 1.

The inverse kinematics can by computed by the relation between rotation matrix and Euler's angle set equation. Equations 15-17 use to find the joint angle that use to command the robot to assume the helical configuration.

$$R_{X'Z'Y'}(\alpha,\beta,\gamma) = \begin{bmatrix} c\beta c\gamma & -s\beta & c\beta s\gamma \\ c\alpha s\beta c\gamma + s\alpha s\gamma & c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma \\ s\alpha s\beta c\gamma - c\alpha s\gamma & s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma \end{bmatrix}$$

$${}^{i-1}_{i}R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} c\beta c\gamma & -s\beta & c\beta s\gamma \\ c\alpha s\beta c\gamma + s\alpha s\gamma & c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma \\ s\alpha s\beta c\gamma - c\alpha s\gamma & s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma \end{bmatrix}$$

$$\alpha = \text{atan2}(r_{32}, r_{22}) \quad (15)$$

$$\beta = a\sin(-r_{12}) \quad (16)$$

$$\gamma = \text{atan2}(r_{13}, r_{11}) \quad (17)$$

After the helical path control is performed, the angle between the non-holonomic constraint of each link and the wheel axis will have the same value. If we change the view point of robot and the pole by spread out the pole surface and robot. we will see that all the vectors that represents non-holonomic constraint are parallel (fig. 5). This process actually keeps the wheel rotational velocity input for each link module to lie within the null space of the non-holonomic constraints (equation 2). Thus, for each module, the dimension of the control input can be reduced to 1 from 4.

The implementation of this proposed controller consists of 3 main parts. The first part is the inner control loop which is the built-in position control of each roll-pitch-yaw motors for all modules. Each motor receives angular position command. The second part is the helical controller that used to transform the helical path into the rotation angle for each motor. The last part is velocity generator that receives the climbing velocity command and transforms it to the desired angular velocity of the robot's wheel. The system diagram can show in fig.6.

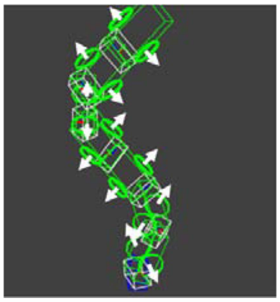

Fig. 5. the robot after apply the result angle and holonomic

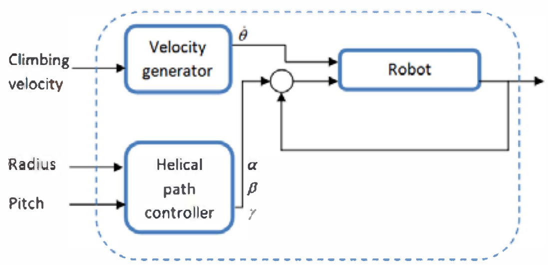

Fig. 6. System diagram.

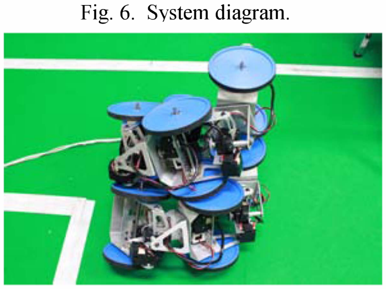

Fig. 7. The prototype snake robot.

The helical climbing of wheeled type is easier to control climbing speed than climbing of non-wheeled type that is needed to control the relation of each joint angle. This robot structure is more complicate and it is necessary for this type of climbing.

VI. EXPERIMENTAL RESULTS

We applied the helical controller to the robot and set up the climbing experiment to confirm the effectiveness of the proposed controller in real situation. The pole that we used in the experiment has 14.5 cm. diameters. The pole was wrapped with 1 mm thick carpet to create the high friction surface. Due to the wheel offset, we used 12.75 cm radius as the input parameter for the helical path for calculating the angle that used to control the robot. At 15 degrees helical pitch angle, the robot can climb the pole at the speed of 1 m/min.

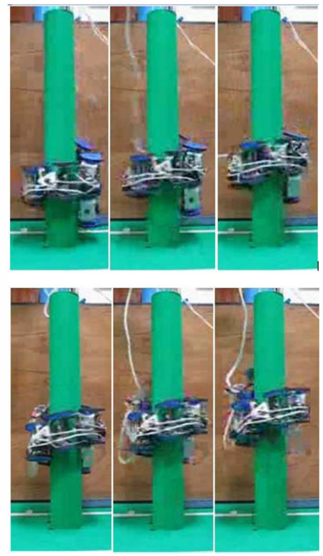

Fig. 8. Climbing experiment of the robot.

VII. CONCLUSIONS

In this paper, we proposed the helical path controller that provides the control solution for a modular wheeled snake robot for pole climbing application. The complexity of the non-holonomic constraints that exist in the modular wheeled snake robot can be reduced by imposing the helical controller to keep the robot in the pose that allow the wheel velocity input of all modules to lie within the null space of the non-holonomic constraints. The helical controller was tested with the real robot and the experiment showed that the robot can successfully climb the pole at the speed of 1m/min.

REFERENCES

[1] Chirikjian, G. S. and Burdick, J. W. "Kinematically optimal hyper-redundant manipulator configurations." *IEEE Transactions on Robotics and Automation 11(1995)*: 794-806.

[2] Kevin Lipkin, Isaac Brown, Aaron Peck, Howie Choset, Justine Rembisz, Philip Gianfortoni, Allison Naaktgeboren, 2007. "Differentiable and Piecewise Differentiable Gaits for Snake Robots", *IEEE/RSJ International Conference on Intelligent Robots and Systems.*

[3] Sean B. Andersson, 2006. "Discrete approximations to continuous curves", *IEEE International Conference on Robotics and Automation.*

[4] Kevin Lipkin, Isaac Brown, Aaron Peck, Howie Choset, Justine Rembisz, Philip Gianfortoni, and Allison Naaktgeboren, 2007. " Differentiable and Piecewise Differentiable Gaits for Snake Robots", *IEEE/RSJ International Conference on Intelligent Robots and System.*

[5] James C. McKenna, David J. Anhalt, Frederick M. Bronson, H. Ben Brown, Michael Schwerin, Elie Shammas, and Howie Choset, 2008. "Toroidal Skin Drive for Snake Robot Locomotion", *IEEE International Conference on Robotics and Automation.*

[6] Seirei industry co.,ltd, 2000, Prunning machine AB232R [Online], Available : http://www.seirei.com/products/fore/ab232r/ab232r.html# [2/11/2009].