Modeling and Gait Design of a 8-Tetrahedron Walker Robot*

Miguel Abrahantes, James Dratz†, Cornelius Smits†, Leif Nelson†
Department of Engineering, Hope College, Holland, MI

***Abstract*— This work describes the use of a simulating walking robot to model to design gaits for a tetrahedron (TET) walker robot. A TET robot has struts and nodes positioned in a way that it allows the robot to move by extending and contracting the struts. The Hope College Controls team has built a working 4-TET robot and has done extensive modeling and gait development on the 8-TET.**

I. INTRODUCTION

Autonomous, reconfigurable robots are becoming important to perform tasks difficult or dangerous to complete for humans [1]. Walking robots are one particular form of autonomous robots, which execute locomotion without wheels. Throughout the development of walking robots [2,3], biological gait patterns have been examined to attain more information on the coordination in these natural gait patterns [4,5], which can be used to design robust and swift walking gaits. The evolution of neural network controllers [7] is also an important part of the development of walking robots.

The Tetrahedron Walker Robots are one class of robotic structures that can be called walking robots. These robotic structures have been developed by a NASA / Goddard Space Flight Center engineering team [7]. These robots utilize a method known as Addressable Reconfigurable Technology (ART). This technique employs the tetrahedron as a unit cell in a single or multiple cell network. Vertices act as nodes to which struts are connected. Conformable tetrahedrons are the simplest space-filling three dimensional forms just like triangles are the simplest plane-filling figures in two dimensional spaces.

Able to function as and when needed to meet mission requirements, ART structures are reusable, reconfigurable, and self-repairing. They are composed of self-similar parts, struts, and nodes, allowing the robots to be assembled in the field into a variety of tetrahedral configurations. A single tetrahedral can travel over irregular terrain. Continuous networks have locomotion with a high degree of freedom resembling amoeboid movement. Different tetrahedral robot configurations are specified by the number and position of tetrahedrons determined by the struts and nodes.

Tetrahedron Walker Robots are more mobile than conventional wheeled or tracked robots. These robots are able to travel across more rugged terrain, when considering the slope, ruggedness and the size of potential obstacles. This

---

† Undergraduate Student

* This work was partially supported by NASA, Grant Number NNX09AH69G and Michigan Space Grant Consortium.

ability allows the robot to evade fatal or injurious falls and helps the robot from becoming trapped permanently. The highly flexible nature of the robot's modularity allows it to conform to almost any landscape it encounters. This capability is possible only if the robot is able to avoid internal and external failure through its gait. Because its structure is made up of simple cells, the robot is easily maintained and capable of performing large amounts of self-repair.

Using more complex n-tetrahedral structures will result higher degrees of freedom. This design uses different levels of intelligences communicated through an interface in order to execute locomotion by smoothly changing the shape of the entire structure. Each tetrahedral cell has to respond in a way that maintains synchronism with the other cells so that binding does not occur in the structure.

II. Simulation of the Robot

We simulated the robot's gait in three dimensional space by using the Simulink™/SimMechanics™ software package. This MATLAB programming library allows users to simulate the dynamics of a model description. The robot model is defined by rigid bodies called links and joints. The links are described by their physical properties such as inertial properties, mass, center of mass, and geometry. Additional parameters include gravity, maximum and minimum joint forces and angles, and collision forces resulting from the robot's links hitting the floor. The allowed joint angles must be restricted manually to prevent the links to intersect each other. The simulation occurs in time steps from time t to time t + d. d is the step of the simulation. Smaller steps allow simulations with greater accuracy, although they slow down the simulation considerably. The control program updates simulation parameters for each time step. These parameters include the rigid bodies' positions, velocities, and accelerations. Once the accelerations are known, numerical integration is employed to determine the bodies' positions and velocities for each time step. Information about the position of the center of mass and of the nodes is saved for later animations.

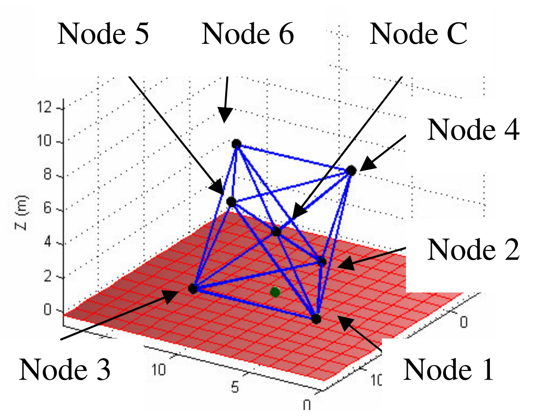


Figure 1: Eight-tetrahedral Structure (8-TET)

III. The Robot

The single-tetrahedral structure is the simplest form of tetrahedral robots. Our focus is on an eight-tetrahedral structure. The tetrahedral structure is overconstrained, because a strut cannot move alone but three or more struts need to move in a coordinated way. This fact makes the control complicated as reliable coordination is essential to prevent binding in the structure. The control requires an accurate definition of the structure's geometry throughout the shape-changing movement of the structure in its walking gait. Our proposed control scheme has the control calculate a complete set of task patterns for each gait or group of gaits before control the robot autonomously. We developed a rolling gait for an 8-TET robot.

A. Rolling gait

The rolling gait has the robot rolling onto another face to execute locomotion. This gait was developed through an in depth analysis of the geometric relationships between the struts in an effort to create a gait that could be executed swiftly and efficiently. The robot goes through a series of steps to complete the gait. The front of the robot is the part facing the new position.

1. Three struts in the back and three struts in the front of the structure extend in coordination to each other.
2. The six struts continue to extend until two faces are on the ground.
3. The three struts in the back continue to expand and the three struts in the front contract. This serves to move the center of mass from the first bottom face to the next bottom face.
4. Now three struts in the back and three struts in the front contract.
5. They continue to contract until the robot is in its original shape.

This amoebic rolling gait allows the robot to travel quickly with decreased wear on the structure due to tumbling.

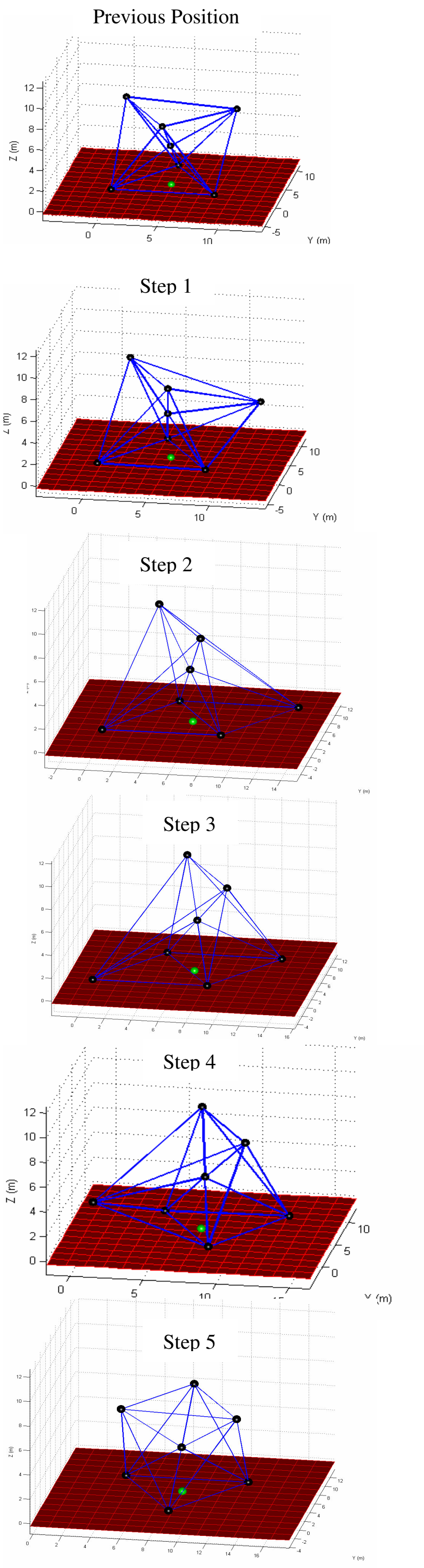


Figure 2: The shape of the 8-TET resulting from each step.

This gait involves six struts and has three gait movements. A gait movement can be defined as the change in the structural shape of the robot when each strut is performing only one type of action (contraction, extension, and others). A sequence of gait movements will define the gait. We have implemented several algorithms to define this gait. S(C,1) is defined as the length of the strut from the center to Node 1. L is the initial length of the strut and x goes from 0 to 1. In the first movement, the struts C-3, 3-5, and 3-6 will be extending with the following length constraints:

$$S(3,C) = L + x * L * (\sqrt{2 - \sqrt{3} * cos(80)} - 1)$$

$$S(3,5) = S(3,6) = S(3,C) * \sqrt{2}$$

The struts C-4, 4-5, and 4-6 will be extending as follows:

$$S(4,C) = L + x * L * (\sqrt{2 - \sqrt{3} * cos(100)} - 1)$$

$$S(4,6) = S(4,5) = S(4,C) * \sqrt{2}$$

In the second movement, the structure moves its center of mass over to its new bottom. To do this, the struts in the back must continue to extend and the struts in the front must contract. C-3, 3-5, and 3-6 will be extending as follows where x goes from 0 to 1:

$$S(3,C) = L + x * L * (\sqrt{2 - \sqrt{3} * c0s(100)} - 1) + (1-x) * L * \sqrt{2 - \sqrt{3} * cos(80)})$$

$$S(3,5) = S(3,6) = S(3,C) * \sqrt{2}$$

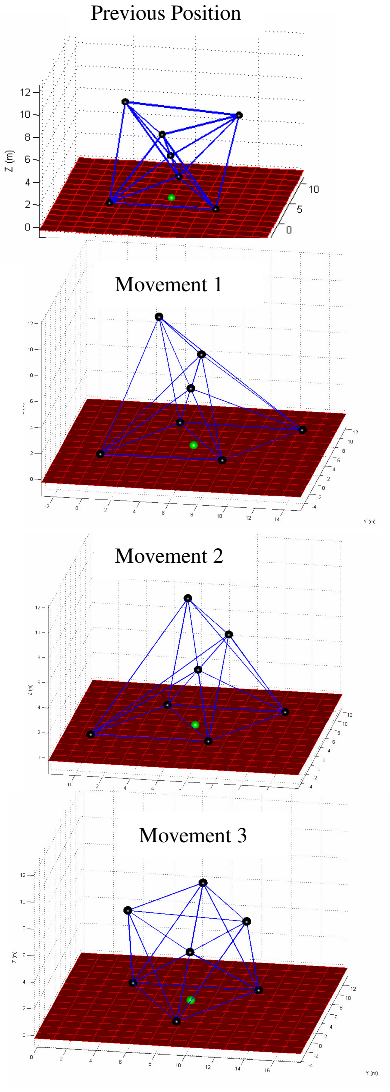
The struts C-4, 4-5, and 4-6 will be contracting as follows:

$$\alpha = cos^{-1}(-\frac{(S(3,C)^2 - 2L^2)}{\sqrt{3} * L^2})$$

$$S(4,C) = \sqrt{2L^2 - \sqrt{3}L^2 * cos(180 - \alpha)}$$

$$S(4,5) = S(4,6) = S(4,C) * \sqrt{2}$$

In the third movement, the robot follows the length constraints for the first movement in reverse.


Previous Position

Movement 1

Movement 2

Movement 3

Figure 3: The shape of the 8-TET resulting from each gait movement.

IV. The Control

There are three levels of controlling for tetrahedral robotic structures. The first commands the struts directly, which becomes impractical when the structure has overstrained struts. The second level controls node position directly. The operator sets the next position for each node individually. The control then calculates the needed strut lengths to reach the next position and executes this motion in a coordinated way. In any tetrahedral structure, there are fewer nodes than struts. It is also easier for the operator to think in terms of node positions rather than strut lengths especially in structures with greater complexity. The third level controls the structure autonomously as it moves through a gait pattern. There are two different ways to generate the gait pattern. In the first type of third level control, one or multiple gait patterns are translated into a set of pre-calculated tasks by the control. This allows the user to control the robot by selecting its next position. The gait patterns can be perfected to increase energy efficiency, speed, and/or stability. With this control, the gait patterns may be only effective in specific conditions. In the second type of third level control, a neural, genetic, or evolutionary algorithm could decide how the robot moves based on the environment around it and the mission objective.

Much of the past research has been focused on understanding gaits (third level controls) or choreographed motions. Since rocks and other types of unpredictable terrain may require a custom gait pattern to be avoided, second level controls are also important to overcome obstacles.

A simulation program using SimMechanics™ has been design to find out new possible gait patterns and save these positions as a series of steps. The program uses interpolation to create a smooth simulation. Once developed, a user can send these commands to a prototype robot, which uses a PID controller with PWM motor control to execute the gait pattern.

V. Conclusion

The ability to simulate the robotic structures allows us to develop new gaits rapidly and analyze them quickly. Simulating the robots is far cheaper than completing experiments with real hardware. The simulation is fast, can be played back and stopped to observe key motions, and does not strain the hardware if an error were to occur. Future work includes the development of new gait patterns and the modeling of more complex tetrahedral structures.

REFERENCES

[1] R.A.Brooks. New approaches to robotics. Science, 253:1227 – 1232, 1991.

[2] M..A.Lewis, A.H.Fagg, and A. Solidum. Genetic programming approach to the construction of a neural network control of a walking robot. In Proceedings of the 1992 IEEE International Conference on Robotics and Automation, pages 2618-2623, Nice, France, May 1992

[3] G.F.Spencer. Automatic generation of program for crawling and walking. In S.Forrest, editor, Proceedings of the 5th International Conference on Genetic Algorithms, ICGA-93, page 654, University of Illinois at Urbana-Champaign, 17-21 July 1993. Morgan Kaufmann.

[4] W. Banzhaf, P. Nordin, R.E.Keller, and F.D.Franconse. Genetic Programming—An introduction: On the Automatic Evolution of Computer Programs and its Applications. Morgan Kaufmann, dpunkt.verlag, 1998.

[5] P. Clark et al (2004) "BEES for ANTS: Space Mission Applications for the Autonomous NanoTechnology Swarm", AIAA ISTC Proc. Chicago, Session 29-IS-13

[6] J.-A.Meyer. Evolutionary approaches to walking and higher-level behaviors in 6-legged animats. In Gomi, editor, Evolutionary Robotics II: From Intelligent Robots to Artificial Life (ER'98) AAAI Books, 1998