

Robust Learning of Tensegrity Robot Control for Locomotion through Form-Finding

Kyunam Kim¹, Adrian K. Agogino², Aliakbar Toghyan¹,
Deaho Moon¹, Laqshya Taneja¹, and Alice M. Agogino¹

Abstract—Robots based on tensegrity structures have the potential to be robust, efficient and adaptable. While traditionally being difficult to control, recent control strategies for ball-shaped tensegrity robots have successfully enabled punctuated rolling, hill-climbing and obstacle climbing. These gains have been made possible through the use of machine learning and physics simulations that allow controls to be “learned” instead of being engineered in a top-down fashion. While effective in simulation, these emergent methods unfortunately give little insight into how to generalize the learned control strategies and evaluate their robustness. These robustness issues are especially important when applied to physical robots as there exists errors with respect to the simulation, which may prevent the physical robot from actually rolling.

This paper describes how the robustness can be addressed in three ways: 1) We present a dynamic relaxation technique that describes the shape of a tensegrity structure given the forces on its cables; 2) We then show how control of a tensegrity robot “ball” for locomotion can be decomposed into finding its shape and then determining the position of the center of mass relative to the supporting polygon for this new shape; 3) Using a multi-step Monte Carlo based learning algorithm, we determine the structural geometry that pushes the center of mass out of the supporting polygon to provide the most robust basic mobility step that can lead to rolling. Combined, these elements will give greater insight into the control process, provide an alternative to the existing physics simulations and offer a greater degree of robustness to bridge the gap between simulation and hardware.

I. INTRODUCTION

Tensegrity robots are an innovative concept based around building a soft, compliant robot based on a tensegrity structure. These tensegrity structures are a unique class of structures constructed by a network of cables connecting isolated rods [1], [2]. The way these structures distribute forces across their members has many analogies to multi-agent systems and networks [3]. Although no rod members touch each other, a tensegrity structure maintains its equilibrium geometry by delicately balancing cable tension and rod compression forces. In other words, rod ends, or *nodes*, where cables are connected, of a tensegrity structure at an equilibrium experience zero net forces. This property of the

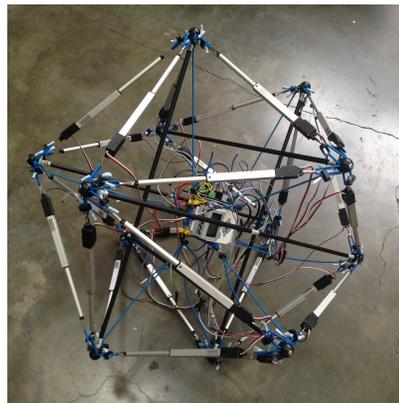


Fig. 1. A six-strut tensegrity robot developed at UC Berkeley. The robot consists of 6 rods and 24 cables. Linear actuators located at the center of each cable are used to deform the robot structure. A controller placed at the center of the structure controls all of the linear actuators.

structure is exploited in Sect. III, where an equilibrium of the structure is found by a dynamic relaxation technique.

Naturally compliant tensegrity structures have several advantages for soft robotic platforms. They are lightweight, robust, energy efficient and capable of a wide range of motions [4], [5], [6], [7]. Moreover, their structural compliance allows them to work beside humans safely. For this reason, tensegrity robots have been envisioned for assistive and rehabilitative healthcare by providing hospital service or direct in-home assistance. Furthermore, U.S. National Aeronautics and Space Administration (NASA) is developing tensegrity robots for space exploration missions [8], [9]. Since multiple tensegrity robots can often be packed into a small volume, NASA has been investigating using multiple cooperative tensegrity robots for planetary missions [10].

However, mobility is required to operate tensegrity robots in such applications. In the literature, simulations as well as physical demonstrations of locomotion of several tensegrity robots have been introduced [11], [12], [13], [14], [15], [16], [17]. These works have shown that different modes of locomotion such as step-wise or punctuated rolling, crawling and undulating are possible with tensegrity robots, depending on their structural geometry. Among many possible tensegrity geometries, this work focuses on a ball-shaped tensegrity robot, especially a six-strut tensegrity robot as its structure has a simpler sphere-like geometry than other tensegrities. This work aims to develop a control for the structural deformation of the tensegrity robot for the purpose of providing mobility to the robot.

¹Kyunam Kim, Aliakbar Toghyan, Deaho Moon, Laqshya Taneja and Alice M. Agogino are with the Department of Mechanical Engineering, University of California at Berkeley, Berkeley, CA 94720 USA {knkim, aliakbar.toghyan, dmoon, laqshya.taneja, agogino}@berkeley.edu

²Adrian K. Agogino is with the University of California at Santa Cruz, Santa Cruz, CA 95064 USA, and also with the NASA Ames Research Center, Moffett Field, CA 94035 USA adrian.k.agogino@nasa.gov

The outline of the work is as follows. A brief description of our tensegrity robot (Fig. 1) is provided in Sect. II. In Sect. III, a form-finding problem of a six-strut tensegrity structure is discussed and a dynamic relaxation technique is applied to find an equilibrium of the structure given an initial structural geometry and imbalanced member forces. In Sect. IV, a Monte Carlo approach for sampling a set of equilibrium configurations of a six-strut tensegrity is described. The sampled configurations are then evaluated based on a fitness function to be defined therein. In order to improve the quality of samples, a learning algorithm which runs multiple generations of Monte Carlo sampling is presented in Sect. V. Then, simulation results obtained in previous sections are tested and validated with our physical robot in Sect. VI. Finally, conclusions are provided in Sect. VII.

II. SIX-STRUT TENSEGRITY ROBOT

Our tensegrity robot (Fig. 1) is based on a six-strut tensegrity structure (also referred to as an expandable octahedron tensegrity) consisting of 6 rigid rods and 24 cables. The structure has 12 nodes and 20 triangles on its outer surface, and the triangles can be categorized into two groups: (a) 8 *closed triangles* each of which is enclosed by 3 cables and (b) 12 *open triangles* each of which is enclosed by only 2 cables. In the original structure, each node is connected to 5 neighbor nodes, 4 of them by tensile members (cables), and the last one by a compressive member (rigid rod). In our robot, however, each cable includes a linear actuator that is used to control the length of the cable. The diagram of edge connection is shown in Fig. 2. The robot is cable-driven and fully actuated, meaning that all 24 cables are independently actuated by linear actuators located at the centers of the edges. Furthermore, the robot has a controller unit as a payload at the center of the structure. Identical fiberglass rods, elastomer cables and linear actuators are used to construct the robot. It is noted here that the connectivity of the members of the robot is unchanged during motion.

The most natural choice of locomotion for this ball-shaped robot is a punctuated rolling motion, or a sequence of *steps* [18]. A six-strut tensegrity robot can realize such motion by repeatedly deforming its geometry by actuating its cables or rods or both [13], [16], [17], [18]. As mentioned previously, however, only cable actuation is of interest in this work.

Depending on the robot's structural geometry, it may or may not succeed in performing a step. Moreover, the desirable geometry will be different depending on the direction of stepping and the surface the robot is standing on at the time of stepping. Consequently, it is crucial to examine which robot geometries realize steps successfully without destroying the structural integrity and exceeding actuation limits of the robot. It is this problem which this work attempts to address. Once the desirable geometry is known, signals can be sent to the actuators to deform the robot structure to match the geometry that will result in a successful step. Furthermore, it is shown in [18] that

the robot can develop different motions, such as forward movement and turning, by repeatedly performing steps.

A condition for a successful step is to place the ground projection of the robot's center of mass (GCoM) outside of its supporting polygon (or its *base triangle*), which can be either an open or a closed triangle. There are three types of steps available for the robot.

- *CO-step* leads the robot from a closed base triangle to an adjacent open base triangle.
- *OC-step* leads the robot from an open base triangle to an adjacent closed base triangle.
- *OO-step* leads the robot from an open base triangle to an adjacent open base triangle.

The robot often performs an OC-step right after a CO-step without pausing. This step will be denoted as a *COC-step*.

Throughout the work, a set of nodal positions of the robot structure will be referred to as a *configuration*. Hence each possible geometric structure has an associated configuration, but not all configurations are realizable because some may violate the rod length constraint or result in cable breakage. If a configuration results in zero net forces at all nodes, then it will be referred to as an *equilibrium configuration*.

A further description of our robot is presented in [18].

III. FORM-FINDING BY DYNAMIC RELAXATION

A. Dynamic Relaxation

The purpose of dynamic relaxation (DR) is to find an equilibrium configuration of a cable net structure in an iterative way, starting from an initial configuration that does not necessarily satisfy the force balance condition at all or a subset of nodes of the structure [19], [20]. External forces applied to the structure may also be considered in the form-finding process. This section describes how the DR with kinetic damping is used to find an equilibrium of a six-strut tensegrity structure. This type of DR has been shown to be stable and convergent for systems with large local disturbances [19], which is the case of tensegrity structures.

The DR with kinetic damping is based on Newton's second law. Consider a node i ($i = 1, \dots, 12$) and assume a force $\mathbf{F}_i(t)$ is applied to the node. Note that the force is a function of time. If we denote the nodal mass as m_i , then the motion of the node is governed by Newton's second law.

$$\mathbf{F}_i(t) = m_i \mathbf{a}_i(t) \quad (1)$$

In the above equation, $\mathbf{a}_i(t)$ is the acceleration of node i at time t . Using the centered finite difference form of the velocity, the acceleration can be approximated.

$$\mathbf{a}_i(t) = \dot{\mathbf{v}}_i(t) \approx \frac{\mathbf{v}_i(t + \Delta t/2) - \mathbf{v}_i(t - \Delta t/2)}{\Delta t} \quad (2)$$

In (2), $\mathbf{v}_i(t)$ is the velocity of node i at time t and Δt is the time difference between two updates.

Substituting (2) into (1) gives an iterative form of velocity update.

$$\mathbf{v}_i(t + \Delta t/2) = \mathbf{v}_i(t - \Delta t/2) + \frac{\Delta t}{m_i} \mathbf{F}_i(t) \quad (3)$$

In (3), m_i has a fictitious value, that is, the mass may or may not be taken from an actual physical system. Usually, m_i and Δt are tuned for good convergence of the algorithm [19]. As a result, if the total force $\mathbf{F}_i(t)$ applied to node i at time t is known, the velocity of the node can be updated for the next time step using (3).

The position of node i at time t , denoted here as $\mathbf{r}_i(t)$, can also be updated using the updated velocity.

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \Delta t/2)\Delta t \quad (4)$$

The DR is an iterative method that aims to find an equilibrium of the structure from an initial configuration which may be given arbitrarily. That is, the initial nodal positions $\mathbf{r}_i(0)$ for all i may be chosen arbitrarily as long as they satisfy the rod length constraint. Furthermore, the initial nodal velocities are set to zero in the DR, that is, $\mathbf{v}_i(0) = \mathbf{0}$ for all i . Because the centered finite difference form is used for the velocity, (3) is slightly modified for the first velocity update.

$$\mathbf{v}_i(\Delta t/2) = \frac{\Delta t}{2m_i} \mathbf{F}_i(0) \quad (5)$$

In summary, if the initial configuration of the structure is known and all of the nodal forces are tracked over time, then the nodal positions, velocities and accelerations can be computed for the later time steps iteratively.

In the DR with kinetic damping, kinetic energy of the system is tracked over time.

$$\text{KE}(t) = \sum_{i=1}^{12} \frac{1}{2} m_i \mathbf{v}_i(t) \cdot \mathbf{v}_i(t) \quad (6)$$

When the peak of the kinetic energy is detected, then all of the nodal velocities and nodal forces are set to zero. This is why the method is called *kinetic damping*. By taking this step, energy is dissipated from the system, moving the system towards a local minimum energy state, or an equilibrium. The iteration restarts from the beginning with the new initial configuration defined as the latest configuration at the energy peak. Finally, the whole process is repeated until the kinetic energy as well as the sum of all nodal force magnitudes converge to zero within an error bound. At convergence, the final configuration is regarded as an equilibrium. An example plot of changes of kinetic energy and the sum of all nodal force magnitudes over time is shown in Fig. 3.

It is noted here that, because of the way the DR finds an equilibrium, the intermediate states do not necessarily represent the actual physical behavior of the structure. Only the final equilibrium configuration is physically meaningful.

B. Nodal Forces

Because each node is connected to a rod and four cables, the nodal force $\mathbf{F}_i(t)$ consists of two different types of forces.

- 1) $\mathbf{F}_i^s(t)$: Spring forces applied by the cables.
- 2) $\mathbf{F}_i^r(t)$: A constraint force applied by the rod.

The total nodal force is then the sum of the two forces.

$$\mathbf{F}_i(t) = \mathbf{F}_i^s(t) + \mathbf{F}_i^r(t) \quad (7)$$

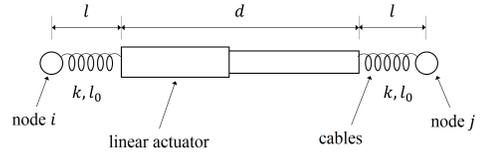


Fig. 2. A diagram of edge connection. Cables are connected to nodes and a linear actuator located at the center of the edge.

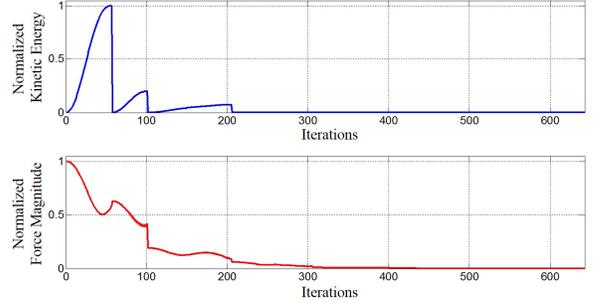


Fig. 3. Changes of kinetic energy and sum of nodal force magnitudes over time during single execution of DR. The quantities are normalized with respect to the maximum values of each case. Notice that both values converge to zero, meaning that an equilibrium configuration is found.

Let $J^i = \{i_1, \dots, i_4\}$ represent a set of neighbor nodes connected to node i by cables. Then, at each time step t , the spring force $\mathbf{F}_i^s(t)$ is the sum of individual cable forces.

$$\mathbf{F}_i^s(t) = \sum_{\forall j \in J^i} \mathbf{F}_{ij}^s(t) \quad (8)$$

In (8), $\mathbf{F}_{ij}^s(t)$ represents the force exerted on node i at time t by a cable connecting nodes i and j . In order to obtain an expression for this force, consider Fig. 2 which depicts an edge configuration of our robot. At each end of the edge are cables connected to nodes, and the two cables are actuated by a linear actuator located at the center of the edge. Because the two cables are identical, when actuated, they stretch the same in lengths. If the cables are linear with stiffness of k and rest length of l_0 , and if the stretched lengths of the cables and the length of the actuator are denoted as l and d , respectively, then $\mathbf{F}_{ij}^s(t)$ has the following expression.

$$\mathbf{F}_{ij}^s(t) = \begin{cases} k(l - l_0) \frac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{\|\mathbf{r}_j(t) - \mathbf{r}_i(t)\|} & \text{if } l > l_0 \\ \mathbf{0} & \text{if } l \leq l_0 \end{cases} \quad (9)$$

$$l = \frac{1}{2} (\|\mathbf{r}_j(t) - \mathbf{r}_i(t)\| - d) \quad (10)$$

In tensegrity structures, cables can only bear tensile forces. Therefore, when their lengths become smaller than their rest lengths, the cable forces are set to zero, as in (9).

In this work, the lengths of the actuators are manipulated to apply forces to the robot structure, and thus, to deform it. Specifically, in Sect. IV, a number of 24-dimensional vectors of the actuator lengths are randomly sampled, and their resultant equilibrium configurations are found and evaluated. It is assumed that the actuators can provide large enough

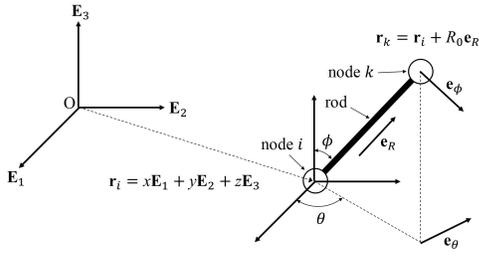


Fig. 4. Coordinate systems used for describing kinematics of two end nodes of a rod. A Cartesian coordinate system, with a set of right-handed orthonormal basis vectors $\{\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3\}$, is used to describe the position of the first node. On the other hand, a spherical coordinate system, with a set of right-handed orthonormal basis vectors $\{\mathbf{e}_R, \mathbf{e}_\phi, \mathbf{e}_\theta\}$, is used to describe the position of the other node relative to the first one. R_0 is the rod length.

forces to fully operate within their stroke range, which is the case of our physical robot.

The next type of force applied to node i is a rod constraint force, $\mathbf{F}_i^r(t)$. This force does not appear in pure tensegrity structures and is a unique feature of tensegrity structures. It is critical to obtain the correct expression for this force, as it guarantees a constant distance between the two end nodes of the rigid rod, and thus, the integrity of the whole structure.

In order to describe the constraint force, the coordinate systems shown in Fig. 4 are used. The position of the first node of a rod is described by a Cartesian coordinate system with a set of right-handed orthonormal basis vectors $\{\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3\}$. Moreover, the position of the other node is described relative to the first one by using a spherical coordinate system with a set of right-handed orthonormal basis vectors $\{\mathbf{e}_R, \mathbf{e}_\phi, \mathbf{e}_\theta\}$. Then, the constraint forces acting on end nodes i and k of the rod are described as follows.

$$\mathbf{F}_i^r(t) = -\mathbf{F}_{ik}^r(t)\mathbf{e}_R(t), \quad \mathbf{F}_k^r(t) = \mathbf{F}_{ik}^r(t)\mathbf{e}_R(t) \quad (11)$$

$$\begin{aligned} \mathbf{F}_{ik}^r(t) = m_k [\ddot{x} \sin(\phi) \cos(\theta) + \ddot{y} \sin(\phi) \sin(\theta) + \ddot{z} \cos(\phi) \\ - R_0 \dot{\phi}^2 - R_0 \dot{\theta}^2 \sin^2(\phi)] - \mathbf{F}_k^s \cdot \mathbf{e}_R \end{aligned} \quad (12)$$

$$\mathbf{e}_R(t) = \frac{\mathbf{r}_k(t) - \mathbf{r}_i(t)}{\|\mathbf{r}_k(t) - \mathbf{r}_i(t)\|} \quad (13)$$

In (12), m_k and R_0 are mass of node k and the rod length, respectively. The coordinates x, y, z, ϕ and θ as well as the force \mathbf{F}_k^s and \mathbf{e}_R are functions of time, but their notations are omitted for better readability of the equation. Because the nodal positions $\mathbf{r}_i(t)$ and $\mathbf{r}_k(t)$ and the nodal velocities $\mathbf{v}_i(t)$ and $\mathbf{v}_k(t)$ are updated iteratively using (3) and (4), the angular coordinates $\phi(t)$ and $\theta(t)$ and their derivatives $\dot{\phi}(t)$ and $\dot{\theta}(t)$ can also be updated using coordinate transformations. Moreover, the acceleration of node i ,

$$\mathbf{a}_i(t) = \dot{\mathbf{v}}_i(t) = \ddot{x}(t)\mathbf{E}_1 + \ddot{y}(t)\mathbf{E}_2 + \ddot{z}(t)\mathbf{E}_3 \quad (14)$$

is already given by (2). As a result, the rod constraint force $\mathbf{F}_i^r(t) = -\mathbf{F}_{ik}^r(t)$ can be updated over time.

Our simulations showed that, when the rod constraint forces are appropriately applied, the maximum error in rod lengths was 0.12% when running the DR.

IV. MONTE CARLO SAMPLING OF EQUILIBRIUM CONFIGURATIONS

In Sect. III, the DR with kinetic damping was used to find an equilibrium of a six-strut tensegrity structure when an initial configuration and a set of actuator lengths were given. Clearly, different sets of actuator lengths will result in different equilibrium configurations. Some of these equilibrium configurations will allow the robot to make a step from one base triangle to another, while the others will not. In order to make a step, GCoM should be placed outside of the robot structure's base triangle. The goal of this section is to find equilibrium configurations that satisfy the preceding condition using a Monte Carlo sampling approach.

A. Sampling

Our cable-driven six-strut tensegrity robot is fully actuated with 24 linear actuators, allowing each cable to be independently actuated. If the target values of 24 actuator lengths are known, then the equilibrium configuration associated with this length set can be found by running the DR.

Let us denote a vector of all 24 actuator lengths as $\mathbf{d} = [d_1, \dots, d_{24}]^T \in \mathbb{R}^{24}$, where d_i is the target length of the i -th actuator. To find desirable equilibrium configurations, a number of instances of the vector \mathbf{d} are sampled by sampling each component d_i independently from a uniform distribution with a physically acceptable range. For each sampled \mathbf{d} , the DR with kinetic damping is performed from an initial configuration of a regular icosahedron, and the resulting equilibrium configuration is found. Each equilibrium is then evaluated with a fitness function described in Sect. IV-B. This process is repeated over a large number of samples. Finally, the best equilibrium configuration and the sampled vector \mathbf{d} that produced this equilibrium are identified.

B. Evaluation

As discussed in Sect. II, in order to make a step from a six-strut tensegrity robot, its GCoM should be placed outside of its base triangle. Assuming masses are uniformly distributed in rods and cables have negligible masses, the center of mass of the structure can be easily obtained once nodal positions at an equilibrium are known.

To evaluate each equilibrium configuration, the following fitness function is used. First, the center of mass is projected onto the planes of the outer surface triangles of the structure. Since, in most cases, the robot steps to and from a closed triangle [18], only this type of triangle is considered when evaluating an equilibrium. That is, the center of mass is projected onto 8 different planes that define the closed triangles. Next, for each projection, the distances between the projected point and 3 edges of a closed triangle are measured, as shown in Fig. 5. If the projected point crosses over an edge and moves outside of a triangle, that distance is measured as a negative value. Because the structural geometry should push GCoM as far as possible from a base triangle for a robust step, our goal is to minimize this distance towards a large negative value. For this reason, 24 distances are computed per equilibrium (3 distances per triangle, 8 closed

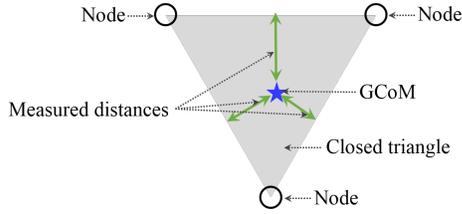


Fig. 5. Distances measured between GCoM and three triangle edges.

triangles) and the minimum of these values is assigned as a score for that configuration. Finally, among a set of equilibrium configurations, the one with the minimum score is picked as the best configuration of the set.

C. Simulation Results

To demonstrate the aforementioned procedure, 5,000 samples of \mathbf{d} were obtained. Their resultant equilibrium configurations were found with the DR and evaluated with the fitness function defined in Sect. IV-B. The physical parameters used in the simulation were taken from our physical robot and are listed in Table I. The result is shown in Fig. 6. The minimum score of all equilibrium configurations was 0.019. Because the lowest score has a positive value, it is anticipated that even this best equilibrium configuration will not allow the robot to make a step as the robot structure's GCoM will stay inside of its base triangle even after structural deformation. To resolve this problem, a learning approach is deployed in Sect. V.

V. LEARNING ALGORITHM

Due to the high dimensionality of sampled vectors \mathbf{d} and the wide interval of the sampling space, it is unlikely that a basic Monte Carlo will discover an adequate solution (Sect. IV). Instead, we use a multi-generation learning algorithm where the highest performing samples from the previous generation are saved and new samples are generated from points that are "close" to the highest performing samples. After enough generations, we expect most of our new samples to come from high performing regions. This learning process can be seen as a multi-step Monte Carlo or as an evolutionary algorithm where at every generation only the "winners" are taken from the previous generation.

A. Multi-Step Monte Carlo

For the samples of the first generation, components of actuator length vector \mathbf{d}_1 , where the subscript denotes its generation, are all sampled from a uniform distribution of the same range, $[d_{min}, d_{max}]$, where d_{min} and d_{max} represent minimum and maximum lengths of actuators, respectively. Once all samples of the first generation are obtained and evaluated according to the fitness function defined in Sect. IV-B, the equilibrium configuration with minimum score, \mathcal{C}_1^* , as well as the actuator length vector sample that produced this equilibrium, \mathbf{d}_1^* , are identified.

For subsequent generations j ($j = 2, 3, \dots$), actuator length vectors, \mathbf{d}_j , are sampled around the best sample

TABLE I
PHYSICAL PARAMETERS OF ROBOT

Parameters	Values
Rod length (R_0)	0.65 (m)
Cable rest length (l_0)	0.038 (m)
Cable stiffness (k)	1193 (N/m)
Minimum actuator length (d_{min})	0.2 (m)
Maximum actuator length (d_{max})	0.3 (m)
Constant offset length (δd)	0.01 (m)

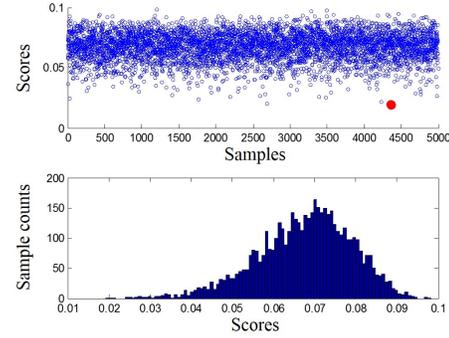


Fig. 6. Scores of 5,000 equilibrium configurations obtained by repeatedly running DR. In the upper figure, the empty blue circles represent scores of all of the equilibrium configurations obtained, and the filled red circle represents the lowest score. The bottom figure is a histogram of the scores of all configurations obtained.

of the previous generation, \mathbf{d}_{j-1}^* . That is, multiple samples of \mathbf{d}_j are obtained from a uniform distribution of $[\mathbf{d}_{j-1}^* - \delta \mathbf{d}, \mathbf{d}_{j-1}^* + \delta \mathbf{d}]$, where $\delta \mathbf{d} = [\delta d, \dots, \delta d]^T$ is a 24-dimensional constant vector. Once the pre-determined number of samples are obtained and evaluated at generation j , the equilibrium configuration with minimum score, \mathcal{C}_j^* , as well as the sampled vector, \mathbf{d}_j^* , producing this equilibrium are found. In generation $(j + 1)$, \mathbf{d}_{j+1} are sampled around \mathbf{d}_j^* in a similar manner, and the process is repeated until termination conditions are met or the pre-defined maximum number of generations is reached.

B. Simulation Results

In our simulation, 30 generations were run with each generation containing 500 samples. Score distributions of chosen generations are shown in Fig. 7. For the early generations, all of the equilibrium configurations found in a single generation had positive scores. However, as generations progressed, the number of samples having negative scores in each generation increased. Moreover, both the minimum and average scores of each generation decreased as the generations evolved (Fig. 8). The figure shows that the first equilibrium configuration with a negative score was found in generation 5, and up to this generation, a total of 2,500 samples were obtained. This was half the number of samples obtained in Sect. IV-C, but the quality of equilibrium configurations turned out to be much better. The minimum and average scores kept decreasing in later generations and were saturated after generation 20. The best equilibrium configuration of all the

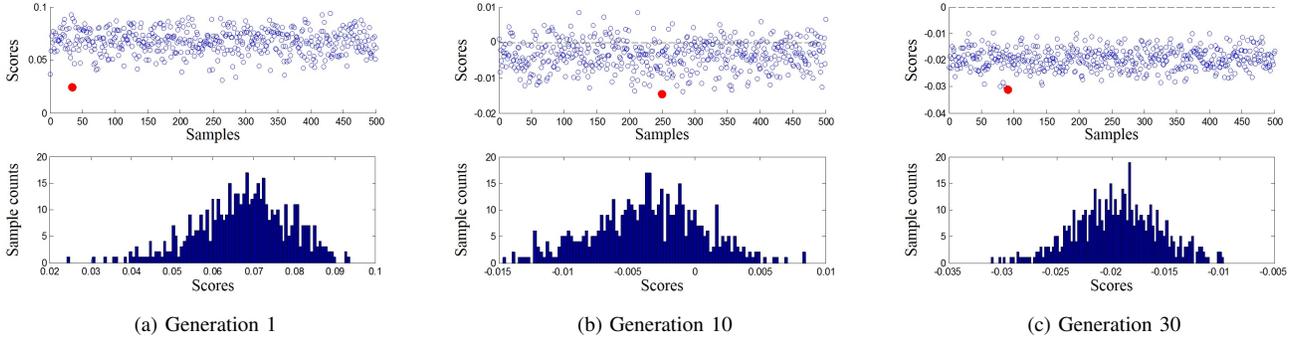


Fig. 7. Score distributions of chosen generations. In the upper figures, empty blue circles represent scores of all of the equilibrium configurations obtained, and the filled red circles represent the lowest scores. The bottom figures are histograms of the scores of all configurations obtained in each generation. 500 equilibrium configurations were obtained in each generation.

TABLE II
ACTUATION POLICY (d_{29}^*)

Actuator edge node pair	(1,5)	(1,6)	(1,9)	(1,11)	(2,7)	(2,8)	(2,9)	(2,11)	(3,5)	(3,6)	(3,10)	(3,12)
Controlled actuator length (m)	0.291	0.300	0.296	0.207	0.200	0.201	0.294	0.298	0.261	0.251	0.297	0.214
Actuator edge node pair	(4,7)	(4,8)	(4,10)	(4,12)	(5,9)	(5,10)	(6,11)	(6,12)	(7,9)	(7,10)	(8,11)	(8,12)
Controlled actuator length (m)	0.251	0.204	0.204	0.291	0.205	0.293	0.285	0.259	0.294	0.299	0.243	0.203

samples was found in generation 29 and -0.031 was its score; this most desirable equilibrium configuration placed its GCoM 0.031m (or 4.77% of the rod length R_0) outside of its base triangle. This configuration is depicted from different orientations in Fig. 11. The set of actuator lengths resulted in this configuration is taken as our actuation policy and is provided in Table II. The node numbers follow Fig. 9.

VI. EXPERIMENTS

A set of experiments was performed with our physical robot (Fig. 1) to test the actuation policy presented in Table II. With this actuation policy, the robot was able to perform either a CO-step or a COC-step, depending on initial actuator lengths. Notice that the methods discussed in the previous sections identify what the most desirable geometry of the robot is in terms of making a step, but they do not provide information on what the intermediate deformations would look like while achieving this geometry from an initial configuration. As a result, it is possible that deforming procedures are different for distinct initial configurations, although their final deformed geometries will be the same. For our tensegrity robot, when the actuators were starting from a fully extended state with initial lengths of d_{max} , the policy in Table II resulted in a CO-step of the robot. However, when the actuators were starting from a half extended state with initial lengths of $\frac{(d_{max}+d_{min})}{2}$, the robot performed a COC-step with the same policy. In the latter case, at the time when the robot made the first CO-step, the robot was still deforming and the width of the landing base triangle was narrow such that GCoM was able to cross over the second base triangle with remaining deformation. Consequently, the following OC-step was automatically performed, and overall a COC-step was made. In contrast, at the moment when the

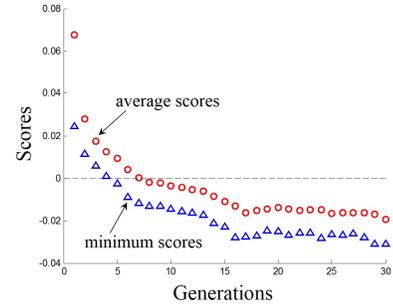


Fig. 8. Learning of minimum (blue triangles) and average (red circles) scores over 30 generations.

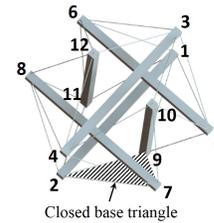


Fig. 9. Node numbers of the robot structure used in Table II.

robot performed the first CO-step in the former case, the width of the landing base triangle had already widened and GCoM was not able to cross over the next base triangle. The trajectories of base triangle nodes as well as GCoM on the ground plane were obtained with a Vicon[®] motion tracking system for these two initial actuator length cases (Fig. 10).

The distance between GCoM and the rotation axis of a base triangle after completion of the deformation was measured as 0.060m which is about twice as large as our estimated value of 0.031m from the simulation. This differ-

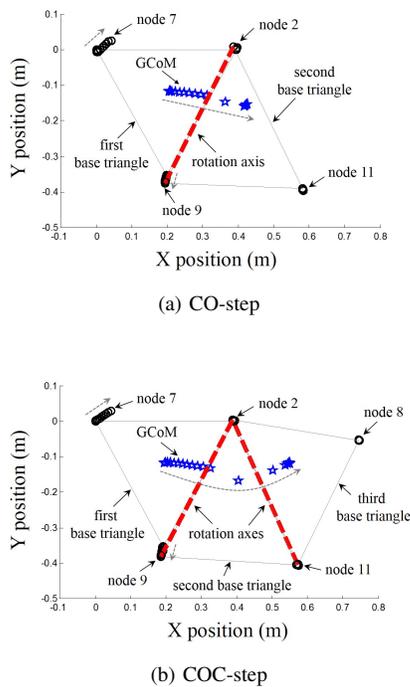


Fig. 10. Trajectories of base triangle nodes and GCoM when the robot structure is deformed with the actuation policy provided in Table II. Depending on initial actuator lengths, the robot performs either a CO-step or a COC-step. In the figures, empty black circles, blue stars and thick red dashed lines represent node positions, GCoMs and rotation axes, respectively. Thin dashed lines show moving directions of nodes and GCoM. Thin gray triangles are base triangles the robot crosses over while performing steps. Node numbers follow Fig. 9. A video of the robot in action can be found at <http://best.berkeley.edu/>.

ence may have caused by the existence of the gravity and the payload on the robot, which was not considered when feasible equilibrium configurations were found in the simulations. When the payload was not placed at the center of mass of the robot, it yielded asymmetry of the structure, causing the robot to favor steps in certain directions. Therefore, in order to properly test the simulated policy on the physical robot, an effort was made to reduce imbalance of the robot structure due to the payload, and the payload was placed as close to the center of mass of the robot as possible. Low resolution linear actuators also contributed to the error as they were not able to deform the robot structure to precisely match the simulated geometry. The resultant deformed structure corresponding to the actuation policy, obtained from both the simulation and physical robot, is presented in Fig. 11 in different views. Furthermore, all of the cable lengths of the deformed structure were measured from both the simulation and hardware robot, and their comparison is given in Table III. Although the lengths match closely in both cases, a maximum of 5.25% error exists.

VII. CONCLUSIONS AND FUTURE RESEARCH

In this paper we show how a dynamic relaxation (DR) algorithm in combination with Monte Carlo sampling can be used to learn robust movements for a ball-shaped tensegrity robot. Not only does this method generate a robotic “step”

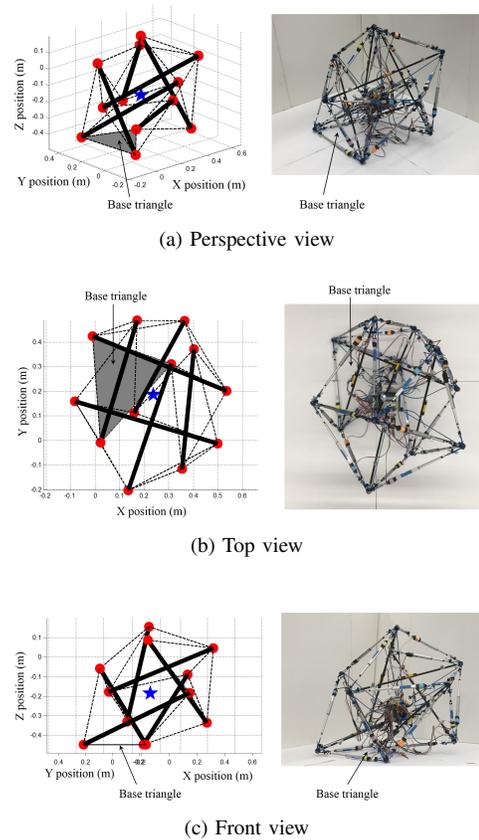


Fig. 11. The equilibrium configuration with the minimum score of all generations. The figures on the left side are simulation results while the figures on the right side show deformation of our physical robot. In the simulation figures, thick black lines, thin dashed lines and red circles represent rods, cables and nodes of the robot structure. Blue stars and gray triangles are center of masses and base triangles of the robot, respectively.

that has a high margin of reliability, the decoupling of form-finding, center of mass computation and actuation decisions allows for more analysis of the learned control algorithm. To achieve these results, we show how the DR can find equilibrium configurations of our six-strut tensegrity robot when its actuator lengths are pre-specified. Then, a number of actuator length vectors are sampled and their resultant equilibrium configurations are found by repeatedly running the DR. The equilibrium configurations obtained are then evaluated based on the fitness function which measures how favorable each configuration is in producing a step. In order to find equilibrium configurations which improve robustness of the step, a multi-generation learning algorithm is used. With this algorithm, the scores of the equilibrium configurations are improved over generations by taking the “winners” of the previous generation to the following generation. From the aforementioned procedure, the actuator length vector which deforms the robot structure in the most desirable way is determined as our actuation policy. The policy is then tested on our physical robot and it allows the robot to successfully achieve a step. In [18], it was shown that the robot can develop different motions by combining multiple steps we call “punctuated rolling” motion.

In our experiments, it was observed that the type of step

TABLE III
EDGE LENGTHS COMPARISON BETWEEN
SIMULATION AND HARDWARE ROBOT

Edge node pairs	Simulated lengths (m)	Hardware lengths (m)	Error (%)
(1,5)	0.431	0.432	0.232
(1,6)	0.433	0.446	3.002
(1,9)	0.425	0.421	0.941
(1,11)	0.342	0.345	0.877
(2,7)	0.340	0.335	1.471
(2,8)	0.366	0.353	3.552
(2,9)	0.432	0.441	2.083
(2,11)	0.443	0.449	1.354
(3,5)	0.417	0.404	3.118
(3,6)	0.381	0.401	5.249
(3,10)	0.434	0.433	0.230
(3,12)	0.363	0.363	0.000
(4,7)	0.398	0.401	0.754
(4,8)	0.364	0.354	2.747
(4,10)	0.335	0.348	3.881
(4,12)	0.436	0.434	0.459
(5,9)	0.336	0.342	1.786
(5,10)	0.434	0.443	2.074
(6,11)	0.423	0.418	1.182
(6,12)	0.399	0.403	1.003
(7,9)	0.433	0.439	1.386
(7,10)	0.437	0.437	0.000
(8,11)	0.397	0.392	1.259
(8,12)	0.346	0.352	1.734

the robot performs is dependent on the initial lengths of actuators even though their final lengths are the same. While our methods can successfully specify the final deformed geometry of the robot for making a step, they are not able to describe the deforming procedures between the initial and final configurations. The authors aim to improve methods presented herein to achieve more consistent steps by further studying deforming procedures.

Furthermore, the fitness function considered in this work only measures how far GCoM moved outside of a base triangle. In terms of energy efficiency, a COC-step is preferred to a CO-step as the robot moves farther with the same actuation policy. For performing of a COC-step, the landing base triangle should not be so wide as to allow GCoM to cross over the second base triangle during the deformation (Fig. 10). For a robust COC-step, this condition may be explicitly added to the definition of the fitness function. Moreover, energy consumption of the linear actuators may also be considered in the fitness function to develop energy efficient actuation policies.

ACKNOWLEDGMENT

The authors gratefully acknowledge support from the NASA Ames Intelligent Robotics Group and partial funding for this research through NASA Early Stage Innovation grant NNX15AD74G, NASA Innovative Advanced Concepts Program and UARC STI Graduate Student Summer Internship. We also thank Prof. Claire Tomlin for lending use of her Vicon[®] motion tracking system, and Borna Dehghani, Ian Kruse and Andrew P. Sabelhaus for their help with the design and construction of the robot.

REFERENCES

- [1] B. Fuller, "Tensegrity," *Portfolio Artnews Annual*, vol. 4, pp. 112–127, 1961.
- [2] A. Pugh, *An introduction to tensegrity*. Univ of California Press, 1976.
- [3] A. Iscen, A. K. Agogino, V. SunSpiral, and K. Tumer, "Robust distributed control of rolling tensegrity robot," in *The Autonomous Robots and Multirobot Systems (ARMS) Workshop at AAMAS*, vol. 2013, 2013.
- [4] R. E. Skelton and C. Sultan, "Controllable tensegrity: A new class of smart structures," in *Smart Structures and Materials '97*. International Society for Optics and Photonics, 1997, pp. 166–177.
- [5] R. E. Skelton, R. Adhikari, J.-P. Pinaud, W. Chan, and J. Helton, "An introduction to the mechanics of tensegrity structures," in *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, vol. 5. IEEE, 2001, pp. 4254–4259.
- [6] R. E. Skelton, "Dynamics and control of tensegrity systems," in *IUTAM Symposium on Vibration Control of Nonlinear Mechanisms and Structures*. Springer, 2005, pp. 309–318.
- [7] R. E. Skelton and M. C. de Oliveira, *Tensegrity systems*. Springer, 2009.
- [8] A. K. Agogino, V. SunSpiral, and D. Atkinson, "Super Ball Bot – structures for planetary landing and exploration," *NASA Innovative Advanced Concepts (NIAC) Program, Final Report*, Jul. 2013.
- [9] V. SunSpiral, G. Gorospe, J. Bruce, A. Iscen, G. Korbel, S. Milam, A. K. Agogino, and D. Atkinson, "Tensegrity based probes for planetary exploration: Entry, descent and landing (EDL) and surface mobility analysis," *International Journal of Planetary Probes*, 2013.
- [10] A. Iscen, "Multiagent learning for locomotion and coordination in tensegrity robotics," Ph.D. dissertation, Oregon State University, May 2014.
- [11] B. T. Mirlletz, I.-W. Park, T. E. Flemons, A. K. Agogino, R. D. Quinn, and V. SunSpiral, "Design and control of modular spine-like tensegrity structures," in *Proceedings of The 6th World Conference of the International Association for Structural Control and Monitoring (6WCSCM)*, Barcelona, Spain, Jul. 2014.
- [12] C. Paul, F. J. Valero-Cuevas, and H. Lipson, "Design and control of tensegrity robots for locomotion," *Robotics, IEEE Transactions on*, vol. 22, no. 5, pp. 944–957, 2006.
- [13] K. Caluwaerts, J. Despraz, A. Işçen, A. P. Sabelhaus, J. Bruce, B. Schrauwen, and V. SunSpiral, "Design and control of compliant tensegrity robots through simulation and hardware validation," *Journal of The Royal Society Interface*, vol. 11, no. 98, 2014. [Online]. Available: <http://rsif.royalsocietypublishing.org/content/11/98/20140520.abstract>
- [14] J. M. Friesen, A. Pogue, T. Bewley, M. de Oliveira, R. E. Skelton, and V. SunSpiral, "A compliant tensegrity robot for exploring duct systems," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, Hong Kong, Jun. 2014.
- [15] J. Bruce, K. Caluwaerts, A. Iscen, A. P. Sabelhaus, and V. SunSpiral, "Design and evolution of a modular tensegrity robot platform," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, 2014.
- [16] A. P. Sabelhaus, J. Bruce, K. Caluwaerts, Y. Chen, D. Lu, Y. Liu, A. K. Agogino, V. SunSpiral, and A. M. Agogino, "Hardware design and testing of SUPERball, a modular tensegrity robot," in *Proceedings of The 6th World Conference of the International Association for Structural Control and Monitoring (6WCSCM)*, Barcelona, Spain, Jul. 2014.
- [17] J. Bruce, A. Sabelhaus, Y. Chen, D. Lu, K. Morse, S. Milam, K. Caluwaerts, A. M. Agogino, and V. SunSpiral, "SUPERball: Exploring tensegrities for planetary probes," in *Proceedings of 12th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS 2014)*, Montreal, Canada, Jun. 2014.
- [18] K. Kim, A. K. Agogino, D. Moon, L. Taneja, A. Toghyan, B. Dehghani, V. SunSpiral, and A. M. Agogino, "Rapid prototyping design and control of tensegrity soft robot for locomotion," in *Proceedings of 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO2014)*, Bali, Indonesia, Dec. 2014, *Finalist Best Student Paper*.
- [19] M. R. Barnes, "Form finding and analysis of tension structures by dynamic relaxation," *International journal of space structures*, vol. 14, no. 2, pp. 89–104, 1999.
- [20] L. Zhang, B. Maurin, and R. Motro, "Form-finding of nonregular tensegrity systems," *Journal of Structural Engineering*, vol. 132, no. 9, pp. 1435–1440, 2006.