Symmetric Reduction of Tensegrity Rover Dynamics for Efficient Data-Driven Control

David Surovik[1] and Kostas Bekris[2]

[1]Deptartment of Computer Science, Rugers University, 110 Frelinghuysen Rd., Piscataway, NJ 08854; david.surovik@rutgers.edu, [2]kostas.bekris@cs.rutgers.edu

ABSTRACT

Tensegrity robots consist of disconnected rods suspended within a network of length-actuated cables, which gives them a high degree of compliance and adaptability suitable for traversing rugged terrain. These vehicles, however, undergo complex contact dynamics that prevent the use of traditional control techniques based on mathematical analyses of equations of motion. Data-driven approaches are thus an appropriate choice for controller design, but are themselves hindered by the high number of degrees of freedom and correspondingly large state spaces.

This paper presents a scheme for exploiting the 24th-order symmetry of an icosahedral tensegrity robot to vastly reduce the breadth of the controller input space without loss of information. Symmetric properties and state-space reduction operations are detailed and placed in the context of a data-driven control pipeline. Results are illustrated by comparing the input and output of a locomotive controller in both raw and symmetry-reduced dynamical spaces. The findings suggest a strong relief of the data requirements for training locomotive controllers.

INTRODUCTION

Tensegrity denotes the design of a truss structure with complete disconnection of its rigid members, such that they appear to float within a network of non-rigid tensile elements such as cables. Due to a lack of rotational constraints at its joints, the structure then responds to forcing by compliantly reconfiguring itself rather than accumulating bending or shear stress. This effectively trades off structural rigidity for resilience, allowing for the use of more light-weight designs. By providing control over the lengths of tensile elements, these structures can be made to actively deform (Skelton and Sultan 1997) and even achieve mobility. The combination of these properties has made tensegrity robots an appealing paradigm for exploration of unstructured terrain, which has motivated the development of NASA's 6-bar spherical tensegrity rover, SUPERball bot (Caluwaerts et al. 2014).

Early work on control of active tensegrities focused on non-mobile platforms via geometric methods (Fest et al. 2004) or through dynamic methods that nonetheless relied upon a consistent interface between the tensegrity and its environment (Aldrich et al. 2003). Analytical control has also been applied to a mobile duct-climbing tensegrity, enabled by specific analysis of its limited and well-behaved contact modes (Friesen et al. 2014). Other mobility behaviors, such as crawling (Shibata et al. 2009), swimming (Bliss et al. 2013), and rolling (Iscen et al. 2015; Zhang et al. 2017) involve more complex contact and friction behavior that is poorly handled by classical control approaches.

The complexities of mobile tensegrities have motivated the use of generic parameterized controllers such as Central Pattern Generators (CPGs) (Bliss et al. 2013) and Artificial Neural Networks (ANNs) (Zhang et al. 2017). CPGs impose structured relationships between various actuators in order to produce rhythmic motions, typically using only a few free parameters. These can be hand-tuned or designed through data-driven approaches such as evolutionary optimization (Paul et al. 2006) that are capable of coping with the highly nonlinear relationship between parameter values and performance. By contrast, ANNs can encode much more complex control policies by incorporating much larger numbers of parameters, whose values are determined through the use of machine learning approaches such as Guided Policy Search (GPS) (Levine and Abbeel 2014).

Recently, GPS has been successfully applied to generate rolling locomotion on the first prototype of SUPERball (Zhang et al. 2017), which features actuation on half of its tensile members. This achievement involved use of the pre-imposed rolling axis that allowed the locomotive gait to be broken into a repeating sequence of six steps. An ANN was produced for each step via independent runs of GPS, and the converged data sets were finally used to initialize training of a single global policy that executes the entire sequence. The forthcoming second prototype of SUPERball will be able to actuate all 24 of its tensile members and thus will not be constrained to a fixed rolling axis. This opens up much broader locomotive possibilities, but also increases the number of viable contact sequences to an extent that cannot reasonably be addressed with the prior approach.

Locomotive vehicles often exhibit physical symmetries that can be leveraged by imposing corresponding symmetric relationships within the control scheme (Valsalam and Miikkulainen 2011), as has been observed to occur in animals (Golubitsky et al. 1999). A related alternative is to map controller outputs onto different actuators at different times, e.g. by using a leg exchange scheme to produce a symmetric biped walking gait (Hyon and Emura 2005). This latter approach presents strong appeal for the SUPERball rover, which exhibits 24th-order symmetry. Better still, such mappings can be applied to controller *inputs* with equal physical validity, allowing their exploitation in data-driven feedback control.

This paper leverages the symmetry properties of the SUPERball rover to design a controller input layer that losslessly reduces the size of the dynamical space visited by a locomotion policy, thus reducing training data requirements. Sec. 2 begins with a generic description for the topology of modular systems and the associated constraints upon maps that exchange identical components. Specific properties and operations for a 6-bar tensegrity are then detailed in Sec. 3, followed by a description of their role and incorporation within a reduced control pipeline in Sec. 4. Finally, a dataset of SUPERball version 2 executing omni-directional rolling is used to illustrate the compactness and improved intelligibility of the reduced dynamical space in contrast to the nominal.

PLATFORM TOPOLOGY

The specific connectivity arrangement for producing a modular robot from sets of many identical components will be referred to as the platform topology. As in the leg exchange scheme of (Hyon and Emura 2005), it is desired to leverage tensegrity sym-

metry by reassigning the identities of specific elements so that information from one context may be reused in some other functionally similar context. This section will introduce generic notation for describing platform topology and element reassignment, along with associated constraints.

Using an index origin of 0, consistent labeling of the various elements of a tensegrity structure is defined by ordering node IDs and element IDs within vectors of potentially different lengths N:

$$\mathbf{n} = \begin{bmatrix} 0 & 1 & 2 & \dots & N_n - 1 \end{bmatrix} \tag{1}$$

$$\mathbf{e} = \begin{bmatrix} 0 & 1 & 2 & \dots & N_e - 1 \end{bmatrix} \tag{2}$$

Variables representing ID values are denoted as $n_i \in \mathbf{n}$ and $e_i \in \mathbf{e}$ respectively; n denotes the node category and e denotes a generic element category. Each category of element will have its own specific instance of a label vector $\mathbf{e}$ and related values, with the set of all element categories denoted $\mathcal{E}$.

The connectivity of a specific element with ID e_i is given by the vector $\mathbf{E}_i$ containing the D_e nodes that it links together:

$$E(e_i) = \mathbf{E}_i = \begin{cases} \begin{bmatrix} n_{e_i 0} & n_{e_i 1} \end{bmatrix} & \text{if } D_e \equiv 2 \\ \begin{bmatrix} n_{e_i 0} & n_{e_i 1} & n_{e_i 2} \end{bmatrix} & \text{if } D_e \equiv 3 \end{cases} \tag{3}$$

Let the label map M be a map that acts upon the node label vector by simply permuting IDs, such that the set of labels is unaltered:

$$M : \mathbf{n} \rightarrow \mathbf{n}' = \begin{bmatrix} n'_0 & n'_1 & n'_2 & \dots & n'_{N-1} \end{bmatrix} \tag{4}$$

$$\{n_i \in \mathbf{n}\} = \{n_i \in \mathbf{n}'\} \tag{5}$$

Overloading notation, let M be applied to elements to ultimately act on the node IDs that define them:

$$M(\mathbf{E}_i) = \begin{bmatrix} M(n_{e_i 0}) & M(n_{e_i 1}) & \cdots \end{bmatrix} \tag{6}$$

Finally, classify M as *topology-preserving* if the set of resulting element vectors is identical to the original set, for each category of element:

$$\{M(\mathbf{E}_i)\} = \{\mathbf{E}_i\} \ \forall \ e \in \mathcal{E} \tag{7}$$

i.e., despite re-labeling of elements, platform topology is unchanged because node connectivity data are equivalent.

The toplogy preservation criterion may be relaxed for systems whose individual elements are themselves symmetric, e.g., if the platform topology can be described as a graph rather than a digraph. This corresponds to an alteration of the ordered node ID tuples of Eq. 3 via operations that describe reorientations of elements:

$$\xi : \begin{bmatrix} n_{e_i 0} & n_{e_i 1} \end{bmatrix} \rightarrow \begin{bmatrix} n_{e_i 1} & n_{e_i 0} \end{bmatrix} \tag{8}$$

$$\xi : \begin{bmatrix} n_{e_i 0} & n_{e_i 1} & n_{e_i 2} \end{bmatrix} \rightarrow \begin{bmatrix} n_{e_i 1} & n_{e_i 2} & n_{e_i 0} \end{bmatrix} \tag{9}$$

$$\chi : \begin{bmatrix} n_{e_i 0} & n_{e_i 1} & n_{e_i 2} \end{bmatrix} \rightarrow \begin{bmatrix} n_{e_i 0} & n_{e_i 2} & n_{e_i 1} \end{bmatrix} \tag{10}$$

Furthermore, consider exponents $v, w \in \{0, 1, 2\}$ to represent the number of times these reorientation operations are applied. The more general criterion for a map M to provide system preservation can then be stated as follows, with per-element reorientation instructions accumulated into vectors $\mathbf{v}$ and $\mathbf{w}$:

$$\exists\ (\mathbf{v}, \mathbf{w}) \in (\mathcal{V}_e, \mathcal{W}_e)\ \mid\ \left\{M\left(\xi^{v,i}\left(\chi^{w,i}\left(\mathbf{E}_i\right)\right)\right)\right\} = \{\mathbf{E}_i\}\ \forall\ e \in \mathcal{E} \tag{11}$$

The reorientation instruction sets $\mathcal{V}_e$ and $\mathcal{W}_e$ can be specified to constrain what is permitted per element category, based on the actual symmetry properties of that type of element.

THE SUPERBALL ROVER

Topology Specifications

NASA's icosahedral tensegrity rover, SUPERball bot, consists of 6 bars $b_i \in \mathbf{b}$ connected at their ends — the nodes $n_i \in \mathbf{n}$ — by a total of 24 cables $c_i \in \mathbf{c}$. In version 1 of the platform, 12 of these cables are actuated by motors while the remaining half are passive; in version 2, with which this paper is exclusively concerned, all 24 cables are actuated. Because the dynamical properties of the bars and cables are symmetric with respect to their endpoint locations, the rotation operation ξ is permissible on an individual basis.

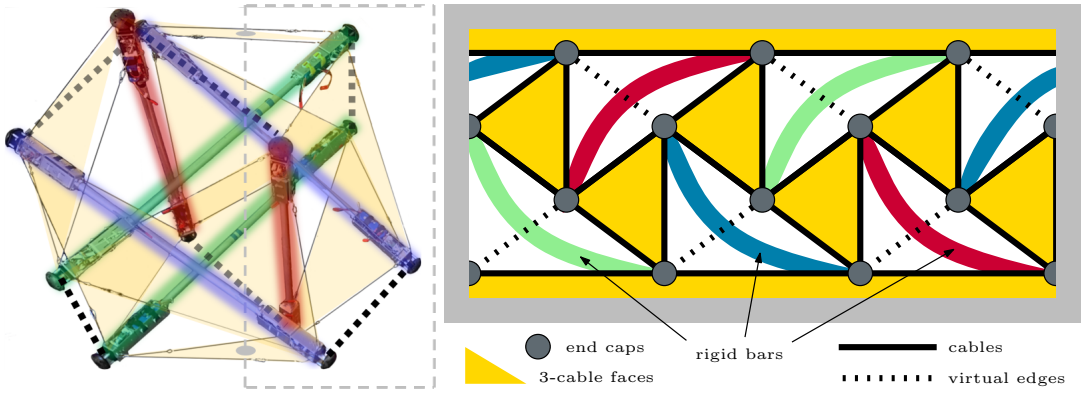

Figure 1. Left: 3D view of SUPERball with colored elements. Dotted black lines signify virtual edges; the dotted grey line defines a cross-section terminating upon the central axis. Right: 2D topological view produced by cutting along the cross-section and unrolling the structure. Straight lines represent elements on the outer surface, while curved lines represent straight elements that span the interior.

The convex hull of SUPERball is composed of 20 faces, 12 of which have cables along two edges while the remainder have cables along all three edges. Rotation of the two-cable category Λ is not permitted, as its edges are not identical in terms of component type, while faces of the three-cable category Δ can be rotated independently. Reflection via χ can be inferred to be relevant on an all-or-none basis with respect to the face elements. The properties and reorientation constraints of all of SUPERball's element categories, $\mathcal{E} = \{b, c, \Lambda, \Delta\}$, are listed in Table 1.

Table 1. Element sets for SUPERball

Category $e \in \mathcal{E}$		Count N_e	Size D_e	Rotations $\mathcal{V}_e$	Reflections $\mathcal{W}_e$
Compressive Bars	b	6	2	$\{0,1\}^N$	n/a
Actuated Cables	c	24	2	$\{0,1\}^N$	n/a
2-cable Faces	Λ	12	3	$\mathbf{0}_N$	$\{\mathbf{0}_N, \mathbf{1}_N\}$
3-cable Faces	Δ	8	3	$\{0,1,2\}^N$	$\{\mathbf{0}_N, \mathbf{1}_N\}$

Symmetry

In standard configuration with all tensile members at equal length, SUPERball is a pseudo-icosahedron, which exhibits pyritohedral symmetry with order $S = 24$. Polyhedrons of this symmetry class may be mapped back onto themselves via any of a set of 24 unique linear transformations L_j involving combinations of rotation and reflection (Holden 1971). These are summarized in Table 2 with I the identity matrix, R_y a rotation about the vertical axis, and F_x, F_y reflections about horizontal and vertical axes respectively.

Table 2. Operations Preserving Pyritohedral Symmetry

	Reference Element	Operation	Expression L	Occurrences
(i)	whole body	Identity	I	1
(ii)		Inversion	$-I$ or $F_x F_y F_z$	1
(iii)	3-cable face	Rotation	$R_y\,(120°)$	8
(iv)		Roto-reflection	$F_y R_y\,(60°)$	8
(v)	2-cable face quartet	Rotation	$R_y\,(180°)$	3
(vi)		Reflection	F_x	3

The spatial transformations L_j can each be associated with a label map M_j that rearranges the platform topology without breaking it. In other words, if $\mathbf{x}_a$ is the configuration of element a in some 3D frame of reference, and L_j transforms it to be equivalent to the original configuration of element b, then M_j relabels element a as b:

$$L_j \mathbf{x}_a = \mathbf{x}'_a = \mathbf{x}_b \quad \Longrightarrow \quad M_j\,(a) = a' = b \tag{12}$$

Crucially, the transformations L_j themselves are of limited use as they do not produce an equivalent configuration for general deformed states of the vehicle; however, the maps M_j can be employed beneficially.

The identity operation (i) is trivial to describe since it represents the nominal element labeling of the platform. For the single map of this category, $M_j\,[n_i] = n_i \;\forall\; i$. One option for computing each remaining node map M_j, as suggested by Eq. 12, is to simply apply L_j to standard node positions $\mathbf{q}_i$ defined relative to the center-of-figure and detect spatial correspondences. To help reveal the nature of the operations, however, subsets of some node maps will be determined below within terms of sketches of relevant reference elements provided in Fig. 2.

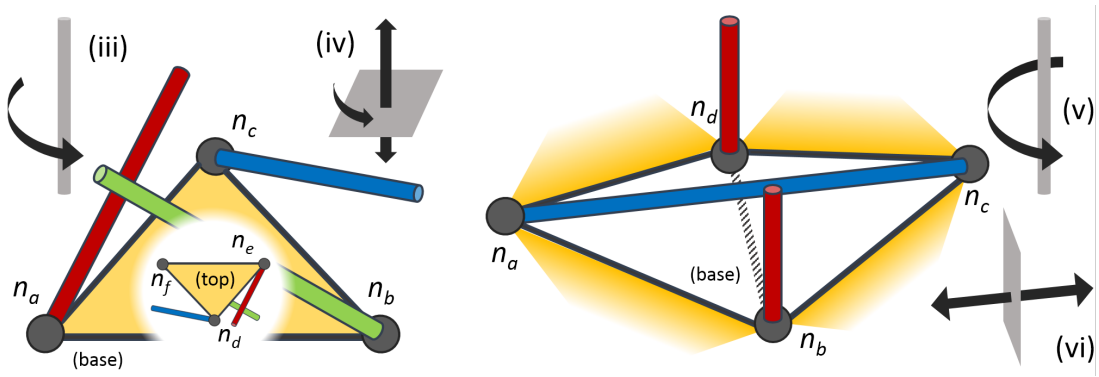

Figure 2. Summary of pyritohedral symmetry operations, shown with respect to applicable elements of SUPERball and labeled in correspondence with Table 2.

Operations (iii) and (iv) can each be applied to each of the eight 3-cable triangles. Using slice notation, the rotation operation (iii) implies $M_j\left[n_a, n_b, n_c\right] = [n_b, n_c, n_a]$. Rotations of -120° are not included as they are equivalent to operation (iii) applied to the opposing 3-cable triangle. Operation (iv) is slightly more complex, involving swapping node IDs with the opposing triangle: $M_j\left[n_a, n_b, n_c\right] = [n_d, n_e, n_f]$. Again, rotations of equivalent magnitude and opposite direction are omitted due to redundancy with applying operation (iv) on the opposing triangle.

Notably, roto-reflection using an angle of 180° actually corresponds to the inversion operation (ii), with $M_j\left[n_a, n_b, n_c\right] = [n_e, n_f, n_d]$. This ultimately results in the same M_j regardless of which of the eight 3-cable triangles it is applied to. Inversion can alternately be posed as reflection along all three axes.

The remaining operations are best described in terms of 2-cable faces, as in the second diagram of Fig. 2. These faces always occur in bordering pairs, and because each operation has an equivalent effect when applied to the opposing face pair, these elements are further grouped into quartets. In the standard configuration of the vehicle, the two virtual edges of each quartet are parallel to each other and perpendicular to those of the remaining quartets. Rotation (v) is realized as $M_j\left[n_a, n_b, n_c, n_d\right] = [n_c, n_d, n_a, n_b]$ while reflection (vi) is simply $M_j\left[n_a, n_b, n_c, n_d\right] = [n_c, n_b, n_a, n_d]$. Rotations or reflections about other axes are identical to the use of operations (v) and (vi) on other 2-cable face quartets.

Altogether, applying the operations of Table. 2 to a reference configuration of SUPERball provides 24 unique topology-preserving maps that relabel structural elements as in Eq. 6 such that Eq. 11 is satisfied.

CONTROL PIPELINE

The key property to be exploited for data efficiency is the dynamical similarity of different state descriptions. This property implies that the system evolves equivalently under some state transformation H and encounters the same operational boundaries, related

by the set $\mathcal{X}$ of all operational states:

$$\begin{aligned} \mathbf{x}_{t+1} = \mathbf{f}\left(\mathbf{x}_t\right) =& H^{-1}\mathbf{f}\left(H\mathbf{x}_t\right) \\ H : \mathcal{X} \rightarrow& \mathcal{X} \end{aligned} \tag{13}$$

Given a set of S unique transformations H_j all satisfying this property, what is then desired is to create a rule for selecting $j\left(\mathbf{x}\right)$ such that all states are mapped into a smaller reference space $\mathcal{X}_R$

$$H_{j(\mathbf{x})}\mathbf{x} \in \mathcal{X}_R \quad \forall\, \mathbf{x} \in \mathcal{X} \quad \text{with} \quad \mathrm{Vol}\left(\mathcal{X}_R\right) = \mathrm{Vol}\left(\mathcal{X}\right)/S \tag{14}$$

with the corresponding reduction of the witnessed phase space sketched in Fig. 3a. Dynamical equivalency allows this to occur without loss of information or accuracy, allowing a control policy to operate solely within the reference space.

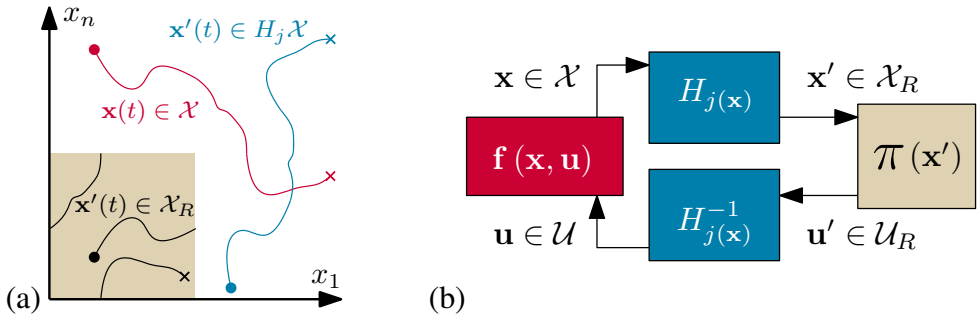

Figure 3. Illustrations of symmetry-based space reduction. (a) Operations H_j preserve dynamical equivalency of state space trajectories; the smaller reference state space $\mathcal{X}_R$ holds an equivalent of all possible trajectories. (b) A control policy π operating solely within the smaller reference spaces.

Construction of the operations H_j for SUPERball is facilitated by the definition of two reference classes of configuration, with visual aid provided in Fig 4. Let the functions $\beta\left(\mathbf{x}\right)$ and $\beta_i\left(\mathbf{x}\right)$ return the category and the ID, respectively, of the triangular face with the lowest center. Using $H_j = M_j$ and defining one triangle of each category as a reference base β_R, the selection of a transformation ID j is restricted to only those options that map the actual base onto the appropriate reference base.

For further utility, let each reference base have its own associated frame, with C_j describing a transformation into this frame from the global frame. C_j consists of a combination of rotations about the vertical axis and reflections along a horizontal axis such that the orientation of the true base face becomes aligned with the reference. Because the global-frame force, gravity, is not affected by C_j, it can be combined with a label map to produce the dynamically invariant body-frame transformation set

$$H_j = M_j C_j \tag{15}$$

As illustrated by the light gray vectors in Fig. 4, however, there are either three or two transformation selections j that map to a reference base triangle. Therefore a

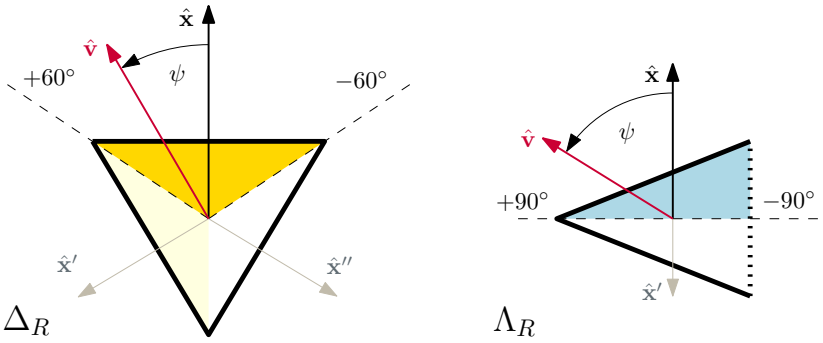

Figure 4. Three-cable and two-cable reference base triangles for SBBv2, with the platform's heading given by $\hat{\mathbf{x}}$ and the slip angle ψ providing a relative description of the velocity direction. Undesired options for transformed headings are shown in light gray.

final key value is used: the slip angle ψ, which denotes the angular distance around the vertical axis from the intended direction of movement to the mean velocity vector of the platform. Imposing the preference of a minimal-magnitude slip angle, a unique symmetric operation can then be selected via

$$\begin{aligned} J_\beta(\mathbf{x}) &= \left\{ j \;\middle|\; \beta(M_j\mathbf{x}) \equiv \beta_R(\mathbf{x}) \right\} \quad \text{with} \quad \beta \in \{\Delta, \Lambda\} \\ j(\mathbf{x}) &= \underset{j \in J_\beta(\mathbf{x})}{\operatorname{argmin}} |\psi(H_j\mathbf{x})| \end{aligned} \tag{16}$$

satisfying the desired state space volume reduction of Eq. 14.

REDUCED TRAJECTORIES

The effect and utility of the reference-space transformation rule of Eq. 16 is now demonstrated using a sample trajectory for axially unconstrained rolling of SUPERball version 2. This trajectory consists of about five full revolutions of the vehicle and three left turns, as conveyed by the top-down view of the center-of-mass path in Fig. 5(a). In complement, Fig. 5(b) shows the paths of half of the nodes relative to the center-of-mass. These smooth and continuous paths exhibit common boundaries and often overlap with each other.

Next, the node trajectory data of Fig. 5(b) is symmetrically reduced using the rule of Eq. 16 and plotted in Fig. 6. The reduction effectively confines each node trajectory to within one or two much smaller bounded regions that are well-isolated from those of other nodes (not pictured: vertical separation). Furthermore, trajectories within each region can be seen to repeatedly trace out similar short segments, clearly signaling the approximate periodicity of the executed behavior. These two traits convey increased density and interpretability of data.

Alternatively, the organization imposed by the symmetric reduction is also apparent in the angular velocity plots of Fig. 7. Only after the reduction can each bar be individually associated with a consistent axis of rotation. Finally, behavioral structure is also made apparent in time series plots of actuator positions, which are the sole control input and also constitute part of the vehicle state. Fig. 8 shows raw trajectory data

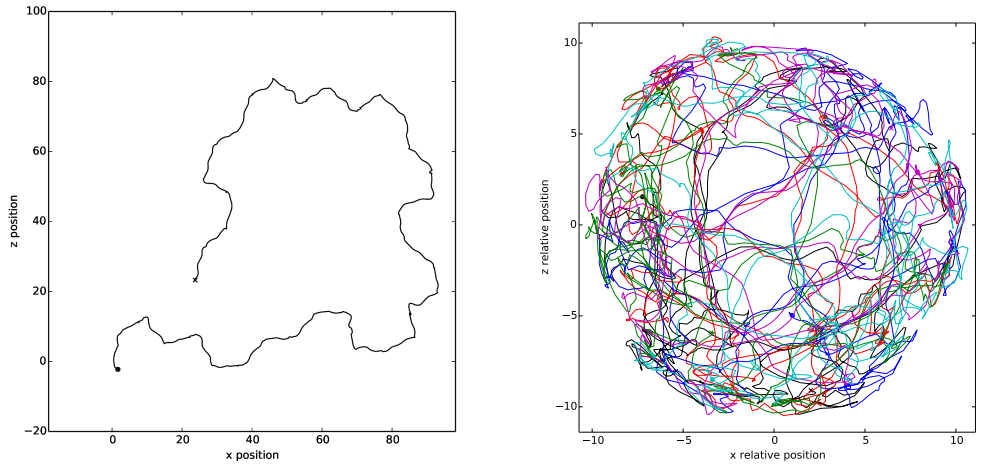

Figure 5. Left: ground-track of platform center-of-mass during execution of rolling and turning trajectory. Right: Center-of-mass-relative ground tracks of six nodes, one per rod.

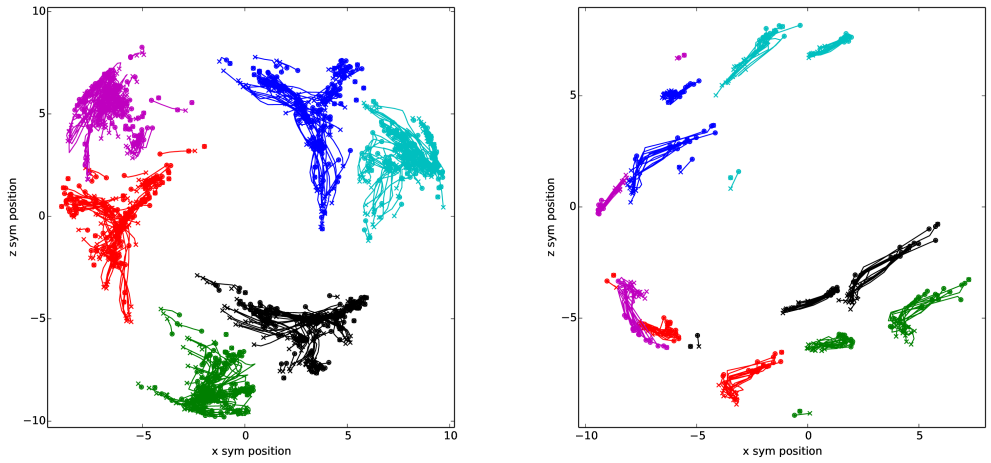

Figure 6. Center-of-mass-relative ground tracks of six nodes, after symmetric reduction, with base triangle type $\beta = \Delta$ (left) and $\beta = \Lambda$ (right).

for six actuators, which exhibit faint and out-of-phase commonalities on the time scale plotted. The symmetry-reduced series of Fig. 9, however, clearly delineates the roles and patterns of different cables. A much faster frequency of periodic behavior is also noted.

CONCLUSION

Many robotic systems exhibit complex dynamics that motivate the use of data-driven techniques for generating effective motion controllers. Although the high dimensionality of tensegrities increases the data burden of such an approach, it is also associated with a high degree of symmetry in the physical structure of the vehicle. By exploiting these symmetric properties, the tractability of data-driven approaches can be greatly improved via a reduction scheme that incurs no approximations or losses, other than to the extent that the components of the physical platform deviate from their intended

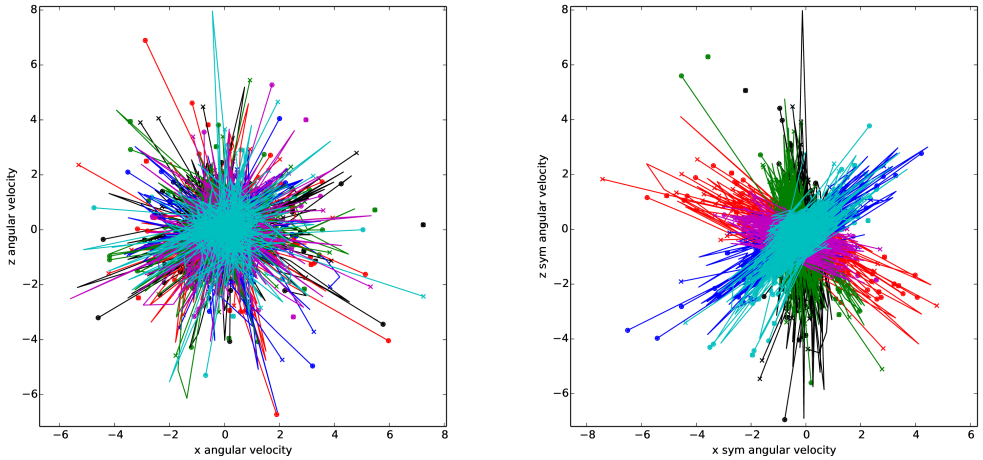

Figure 7. Ground-plane components of angular velocity for all six bars. Left: raw data. Right: symmetry-reduced data.

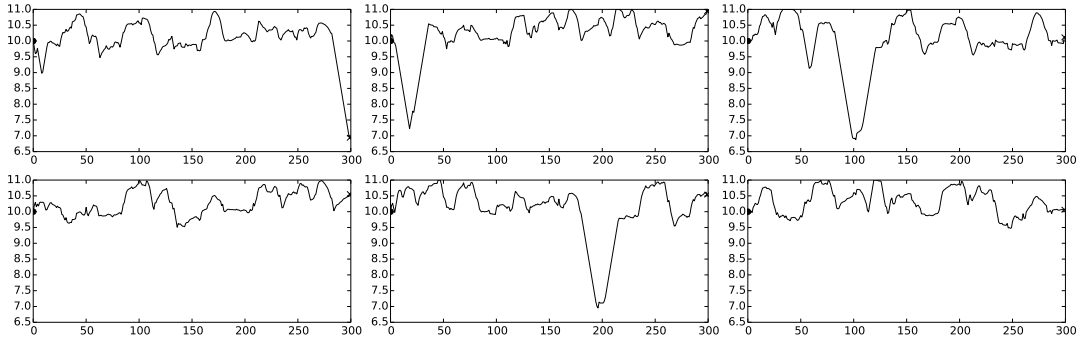

Figure 8. Raw time-varying lengths of the first six actuated cables.

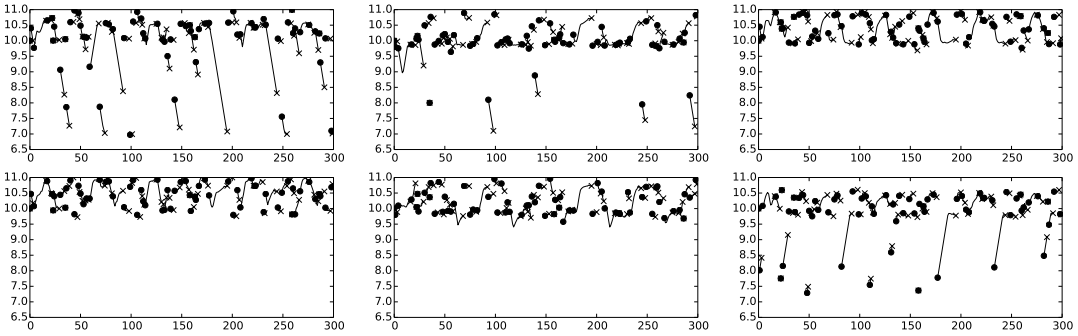

Figure 9. Symmetry-reduced time-varying lengths of the first six actuated cables.

uniformity. While the included demonstrations have shown the effectiveness of the method for a spherical tensegrity rover, it is anticipated that adaptation to other symmetric modular systems and multi-legged vehicles could be easily achieved.

REFERENCES

Aldrich, J. B., Skelton, R. E., and Delgado, K. K. (2003). "Control Synthesis for Light and Agile Robotic Tensegrity Structures." *ACC*.

Bliss, T., Iwasaki, T., and Bart-Smith, H. (2013). "Central Pattern Generator Control of a Tensegrity Swimmer." *Trans. on Mech.*, 18(2).

Caluwaerts, K., Despraz, J., Işçen, A., Sabelhaus, A. P., Bruce, J., et al. (2014). "Design and control of compliant tensegrity robots through simulation and hardware validation." *JRSI*, 11(98).

Fest, E., Shea, K., Smith, I. F. C., and Asce, M. (2004). "Active Tensegrity Structure." *JSE*, 130(October), 1454–1465.

Friesen, J., Pogue, A., Bewley, T., de Oliveira, M. C., Skelton, R. E., and SunSpiral, V. (2014). "DuCTT: A tensegrity robot for exploring duct systems." *ICRA*, 4222–4228 (May).

Golubitsky, M., Stewart, I., Buono, P.-L., and Collins, J. J. (1999). "Symmetry in locomotor central pattern generators and animal gaits." *Nature; London*, 401(6754), 693–5.

Holden, A. (1971). *Shapes, space, and symmetry*. Courier Corporation.

Hyon, S. and Emura, T. (2005). "Symmetric Walking Control: Invariance and Global Stability." *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 1443–1450 (April).

Iscen, A., Caluwaerts, K., Bruce, J., Agogino, A., SunSpiral, V., and Tumer, K. (2015). "Learning Tensegrity Locomotion Using Open-Loop Control Signals and Coevolutionary Algorithms." *Artificial Life*, 21(2), 119–140.

Levine, S. and Abbeel, P. (2014). "Learning neural network policies with guided policy search under unknown dynamics." *Advances in Neural Information Processing Systems*, 1071–1079.

Paul, C., Valero-Cuevas, F. J., and Lipson, H. (2006). "Design and control of tensegrity robots for locomotion." *IEEE Transactions on Robotics*, 22(5), 944–957.

Shibata, M., Saijyo, F., and Hirai, S. (2009). "Crawling by body deformation of tensegrity structure robots." *ICRA*.

Skelton, R. E. and Sultan, C. (1997). "Controllable Tensegrity: A New Class of Smart Structures." *Proc. of the SPIE*.

Valsalam, V. K. and Miikkulainen, R. (2011). "Evolving symmetry for modular system design." *IEEE Transactions on Evolutionary Computation*, 15(3), 368–386.

Zhang, M., Geng, X., Bruce, J., Caluwaerts, K., Vespignani, M., SunSpiral, V., Abbeel, P., and Levine, S. (2017). "Deep reinforcement learning for tensegrity robot locomotion." *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 634–641 (May).