The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Generating Gaits for Snake Robots by Annealed Chain Fitting and Keyframe Wave Extraction

Ross L. Hatton and Howie Choset
{rlhatton, choset}@cmu.edu

***Abstract*— Snake robots have many degrees of freedom, which makes them both extremely versatile and complex to control. In this paper, we address this complexity by introducing two algorithms. *Annealed chain fitting* efficiently maps a continuous backbone curve to a set of joint angles for a snake robot. *Keyframe wave extraction* takes joint angles fit to a sequence of backbone curves, and identifies parameterized periodic functions which produce those sequences. Together, they allow a designer to conceive a gait in terms three-dimensional shapes and translate them into easily manipulated wave functions. We validate the algorithms by using them to produce rolling gaits for crawling and climbing.**

I. Introduction

Snake robots are mobile actuated chains that can locomote via coordinated flexing of their bodies. Typically, snake robots possess many active joints, giving them great versatility, but also making it infeasible for an operator to individually control each degree of freedom. A common solution to this difficulty is to define parameterized *gaits*, or patterned joint motions, and have the operator select among them to direct the robot. The key question then becomes how to design useful gaits.

The gait design approach taken in this paper is based on our observation that gait designers think more easily in terms of the three-dimensional shape assumed by a snake robot than in terms of joint angles. With this approach, we first specify a gait as a moving three-dimensional curve, then discretize it in time. From this discretization, we extract a series of keyframe backbone curves that the robot must pass through in order to execute the gaits. For each backbone curve, we then find the set of joint angles that fits the robot to the curve. We then reassemble these sets of joint angles into numerical functions describing the trajectory of each joint angle. Finally, we perform a second fitting operation, which identifies analytical functions that minimally capture the form of the numerical functions.

II. Background

Snake robots have been studied since at least 1971, with Hirose's pioneering work on the Active Cord Mechanism (ACM) [1]. While this and other early snake robots were confined to planar motion, much recent effort has been directed towards mechanisms that can assume full three-dimensional shapes. Notable developments in this area include Yim's Polybot [2], Mori and Hirose's ACM-R3 [3], our modular snakes ("modsnakes"), shown in Fig. 1 [4], Gonzalez-Gomez *et al.*'s Hypercube [5], [6], and Goldman and Hong's HyDRAS [7], [8].

Fig. 1: Our modular snake ("modsnake") robots have sixteen joints, arranged to allow full workspace flexibility. In past work, we have demonstrated a variety of successful motion strategies for these robots, including traversal of both smooth and rough terrain, swimming, and pole climbing (shown here).

Gaits are commonly defined for snake robots by specifying the joint angles as oscillating functions of time. Hirose's *serpenoid curve* sets the joint angles sinusoidally along the body of the robot and varies the phase of the sine wave in time to generate locomotion [1], [9]. Chirikjian and Burdick [10] extended this principle to a wider range of waves and locomotion modes. We have found much success in applying a three-dimensional variation of this approach to generate crawling, climbing, and swimming gaits [11], [12]; similar results have been demonstrated by Gonzalez-Gomez, *et al.* [5], [6], using central pattern generators (CPGs) to generate the waves in a distributed fashion. The use of CPG-generated waves is becoming increasingly widespread in locomotion control, and is thoroughly reviewed in [13].

Several researchers have applied forms of backbone fitting to find the basic waves that make up snake robot gaits. Chirikjian and Burdick [14] analyzed the curvature of a sidewinding snake, Andersson explored algorithms for fitting a chain of universal joints to a continuous curve [15], and Goldman and Hong [7], [8] examined the kinematics of climbing a pole. These approaches share the limitation, however, of assuming either planar mechanisms or omni-directional flexibility at each joint.

III. Gait Design for Modsnake Robots

Our modsnake robots are constructed of chains of single-degree-of-freedom modules. Each module consists of a joint and a link, and the joint axis of each module is rotated around

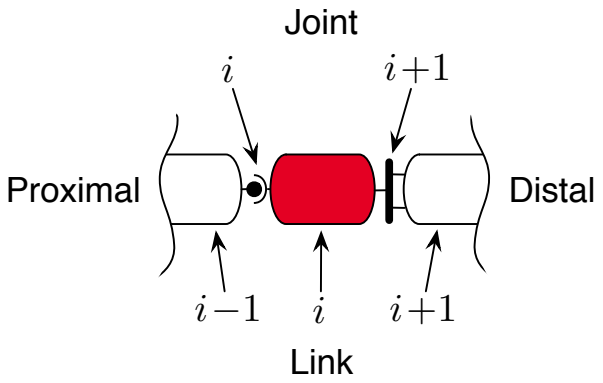


Fig. 2: Modsnake geometry. The ith link is distal to the ith joint, and together they make up the ith module. Each joint is rotated around the central axis of the snake by 90° with respect to the previous joint; in this illustration, joint i generates rotation in the plane, while joint $i+1$ generates rotation out of the plane.

the central axis of the snake by 90° with respect to the previous joint, as illustrated in Fig. 2 [4]. This geometry allows full three-dimensional flexibility while maintaining smaller individual links and greater robustness than equivalent designs combining pairs of modules into active universal joints. We use the convention that the ith link is distal to the ith joint, and that the angle of the ith joint thus determines the position of link i with respect to link $i-1$.

We control these robots by designing gaits which propagate waves along their bodies [12]. The first stage in this wave-based approach is to divide the modules into odd and even sets by joint number, such that all the joint axes in each set are parallel when the snake robot is stretched out. We then take parameterized, periodic functions in time and joint number for each set, and manipulate those parameters to design useful gaits. For example, a basic form of this approach is to define $\alpha(n,t)$, the angle of the nth joint at time t, as

$$\alpha(n,t) = \begin{cases} \beta_{\text{odd}} + A_{\text{odd}} \sin(\theta_{\text{odd}}) & \text{odd} \\ \beta_{\text{even}} + A_{\text{even}} \sin(\theta_{\text{even}} + \delta) & \text{even} \end{cases} \quad (1)$$

$$\theta_{\text{odd,even}} = (\Omega_{\text{odd,even}} n + \omega_{\text{odd,even}} t)\,, \quad (2)$$

where β, A, θ, and δ are respectively offset, amplitude, frequency, and phase shift. The parameter Ω describes the spatial frequency of the macroscopic shape of the robot and the temporal component ω determines the frequency of the actuator cycles. This single wave model encompasses a wide variety of gaits, from slithering and sidewinding to the rolling helix used to climb the pole in Fig. 1, and is similar to the parameterized CPG functions used in [5], [6].

This wave-based approach is a powerful tool, as it greatly reduces the complexity of designing gaits. Rather than individually choosing a trajectory for each joint angle, the gait designer can work with a smaller set of parameters that apply across all the joints. It is not a complete solution, though, as it still leaves open the question of how to select regions of this parameter space which correspond to useful gait types.

To date, our efforts in this direction, while fruitful, have been largely empirical. For instance, through experimentation we have determined that an offset of $\delta = \frac{\pi}{4}$ can be used to generate a sidewinding motion, while an offset of $\delta = \frac{\pi}{2}$ corresponds to rolling motions [12]. Unfortunately, this empirical approach is limited in that it is much easier for a gait designer to think in terms of *backbone curves* [14], *i.e.*, the three-dimensional shapes assumed by the robot, than in terms of wave parameters. Without considerable experience, it is hard to translate the former into the latter, especially if the snake is convoluted enough that the "odd" and "even" joints no longer correspond to "horizontal" and "vertical" bending. It is even harder to identify when a conceived gait cannot be realized via the wave model in (1), and then to generate a new wave function with which to express it.

IV. Annealed Chain Fitting

As translating from a three-dimensional backbone curve to a set of joint angles is difficult for a gait designer, the first stage of our approach automates this operation. For this purpose, we take a given backbone curve as being a three-dimensional locus, along with a "twist" value ϕ that specifies the roll angle of the first link of the robot with respect to the curve's tangent. While in principle it would be possible to solve directly for the set of joint angles which best fit the robot to the locus, such an operation would be computationally prohibitive and, with no way to generate a good starting guess, prone to getting stuck in suboptimal local minima. Instead, we take an iterative approach, and progressively "sculpt" the robot onto the backbone in a series of smaller, more stable optimizations. If the robot had spherical or universal joints, this sculpting process would be trivially accomplished by sequentially setting each joint angle to place the corresponding link directly on the curve. With our alternating-axis geometry, however, the alignment of a given joint axis with the curvature of the backbone (and thus the ability of that module to fit to the curve) depends on the joint angles of the preceding modules. Because our fitting algorithm must consequently allow for the adjustment of previously fit joint angles as each new module is considered, we refer to this process as *annealed chain fitting*.

At each iteration of the algorithm, we separate the joints of the snake robot into three categories: *fixed joints*, which have been previously fit to the curve and will not be changed; *active joints*, which are being fit to the curve; and *free joints*, which will be fit to the curve in future steps. As shown in Fig. 3, the joints are assigned to the categories via a moving *active window*. The size of this window determines the number of active joints, and the step size controls how many active joints are fixed (and how many free joints are made active) between each iteration.

Once the active modules have been selected for an iteration, their joint angles are optimized to fit the modules as closely as possible to the backbone curve. In this paper, we take the sum of squared distances of the distal ends of the modules from the backbone as our objective function, with modifications to penalize sharp angles and "bunching up" along the curve. This objective function essentially creates "sliding springs" which pull the modules towards the

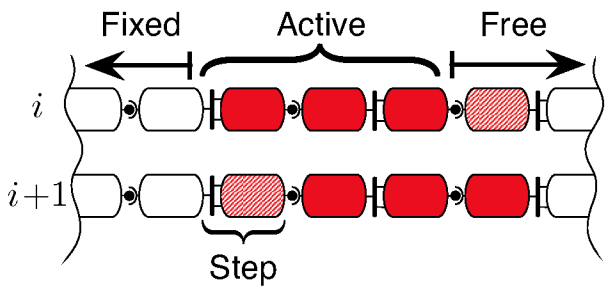


Fig. 3: Annealed chain fitting, with a window size of three and a step size of one. In each iteration, the active window steps along the snake robot. As the step size illustrated here is one module, one free module becomes active and one active module becomes fixed between iterations i and $i+1$.

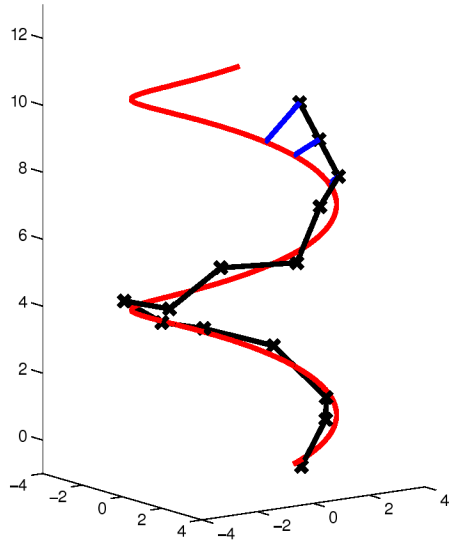


Fig. 4: Fitting a Modsnake to a helix. The active modules, at the top of the plot, are pulled towards the backbone by the attraction lines connecting them to the curve. The fixed modules have already been fit to the curve; their alternating axes of flexion lead to the characteristic zigzag pattern.

curve, as in Fig. 4. This sliding action means that the point on the backbone that attracts each module varies over the optimization, so we do not attempt to analytically solve for the best fit and instead use a numeric solver.

Careful consideration of the active window and step sizes is required. For an N-module robot, with active window and step sizes a and s respectively, the annealed chain fitting algorithm requires searching $\frac{N-a}{s}$ a-dimensional spaces. A large value of a increases the flexibility of the active region, and hence its ability to fit to the backbone. However, this benefit comes at the cost of exponentially increasing the size of each search space, while only linearly decreasing the number of searches. Similarly, large values of s reduce the number of searches but also result in several modules being added in each iteration, which can trap the system in undesirable local minima of the objective function, such as bridging over the coils of a helix. Also, too large a ratio of $\frac{s}{a}$ reduces the number of iterations in which a given module is allowed to move, and thus reduces the "annealing" properties of the algorithm. Given these considerations, and observed diminishing returns in fit quality for $a > 3$, we used values of $a = 3$ and $s = 1$ for the examples in this paper.

V. Keyframe Wave Extraction

For any given gait, we can find a discrete function $\alpha = g(n,t)$ that describes the joint angles for that gait by borrowing the notion of key frames from the animation community and discretizing that gait into a sequence of keyframe backbone curves at a series of times t. We can then use the chain fitter to discretize each backbone into a set of joint angles. If our goal were only to "play back" the gait on a modsnake robot, then simply finding g would be sufficient. However, our goal is deeper; *we want to identify a parameterized function which describes not only the discretized gait, but also all gaits that are qualitatively similar to it.* To this end, we separate out the fitted angles into their odd and even subsets, and then plot these subsets discrete functions

$$\alpha_{\text{odd}} = g_{\text{odd}}(n_{\text{odd}}, t) \tag{3}$$

and

$$\alpha_{\text{even}} = g_{\text{even}}(n_{\text{even}}, t). \tag{4}$$

By observing patterns in these functions, we can identify simple parameterizable waveforms f_{odd} and f_{even} which evaluate to g_{odd} and g_{even} at their respective integer values.

A. Rolling in an Arc

For instance, consider the rolling gait illustrated in Fig. 5. In this gait, the snake robot forms an arc in the plane of the ground and rolls laterally. To find the joint angles which form this gait, we chain fit the modsnake to the arc at a full revolution's worth of twist angles, corresponding to the shapes shown in Fig. 5 and their intermediate positions.

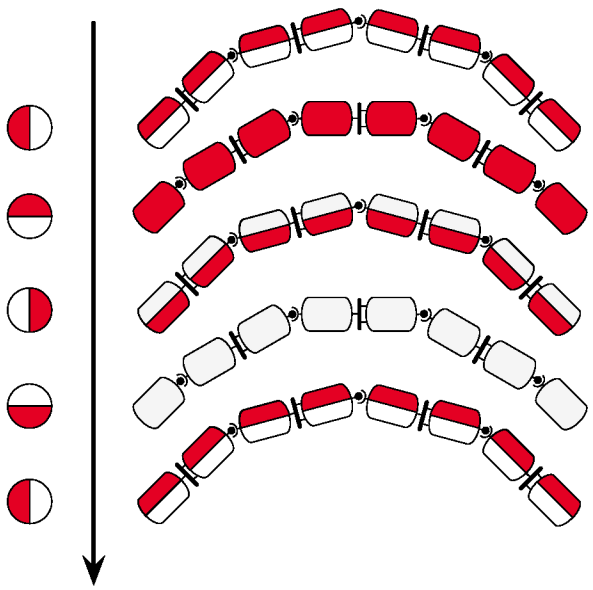
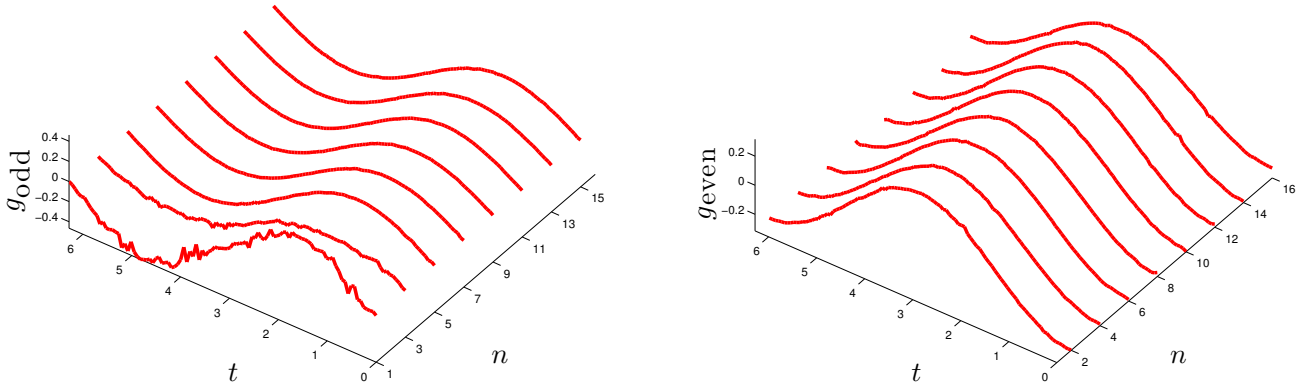
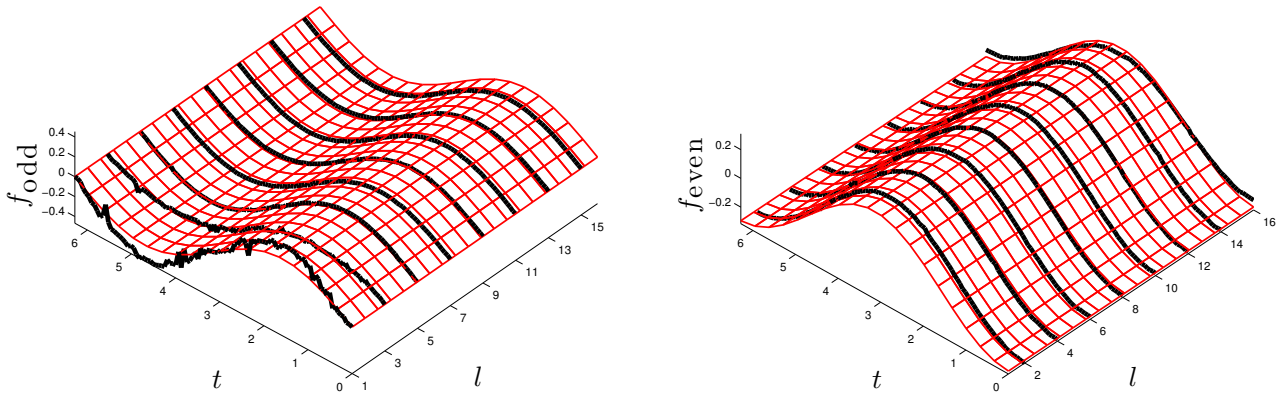


Fig. 5: Rolling in an arc. In our basic rolling gait, the modsnake forms an arc in the plane of the ground and rolls laterally. As the snake rolls, the joints vary between being perpendicular and parallel to the ground plane. At left, an end-view of the modules illustrates the wheel-like motion.



(a) Joint angles of the nth modules from chain fitting to a rolling arc, as in Fig. 5.



(b) Standing waves which generate the joint angles for the rolling arc.

Fig. 6: Waveform for a rolling arc gait. As in Fig. 5, at each time t, the first link of the modsnake is made tangent to the arc with a twist angle of $\phi = t$, and the remaining links are fit to the curve. With the exception of the first two odd modules (which are distorted by boundary condition effects) there is no variation in the joint angles with respect to module number, and sinusoidal variation with respect to time. These joint angles are described by a pair of standing waves of equal magnitude which are phaseshifted from each other by $\delta = \pi/2$.

Extracting the individual joint angles from the fitting data results in the plots of g_{odd} and g_{even} in Fig. 6(a).

The plots in Fig. 6(a) are line plots, as the time discretization can be made small enough to approach continuity, but the spatial resolution is fundamentally limited to the actual modules of the robot. Although the first two odd modules are slightly distorted by boundary condition effects, there are clear patterns to g_{odd} and g_{even}. Each is sinusoidal in time and shows no variation with respect to module number. As there is a quarter-period offset in time between the odd and even functions and they have equal magnitudes, we can assign a pair of continuous functions of the form

$$f_{\text{odd}}(l,t) = A\sin(\omega t) \tag{5}$$

$$f_{\text{even}}(l,t) = A\sin(\omega t - \frac{\pi}{2}) \tag{6}$$

to the gait, which, as in Fig. 6(b), are equal to g_{odd} and g_{even} at $l = n_{\text{odd}}$ and $l = n_{\text{even}}$, respectively. This waveform matches our previous results for generating rolling gaits [11], where we experimentally determined that restrictions of (1) with $A_{\text{odd}} = A_{\text{even}}$, $\beta_{\text{odd}} = \beta_{\text{even}} = 0$, $\Omega_{\text{odd}} = \Omega_{\text{even}} = 0$, $\omega_{\text{odd}} = \omega_{\text{even}}$ and $\delta = -\frac{\pi}{2}$ form the snake robot into the rolling arc shown in Fig. 7.

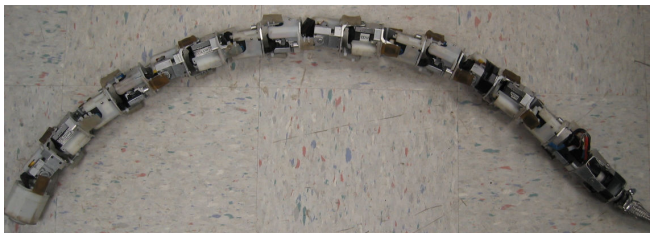


Fig. 7: Modsnake in its rolling arc configuration

B. Rolling in a Helix

One of the more intriguing features of a snake robot is its ability to climb a pole by wrapping around it in a helix, as shown in Fig. 1, and then rolling up the pole. In this gait, as illustrated in Fig. 9, each segment of the body rolls along the pole, effectively acting as a wheel. As with the arc gait, this rolling motion is entirely driven by flexure, with no rotary joints aligned with the direction of rolling. We have previously used our empirical approach to find one such gait for our modsnakes. Others have found an analytical solution for a rolling helix gait for a universal-joint snake robot [7], [8], but it relies on both the bidirectional flexibility of the universal joints and the symmetries inherent in rolling a robot with a cylindrical cross-section up a cylindrical pole,

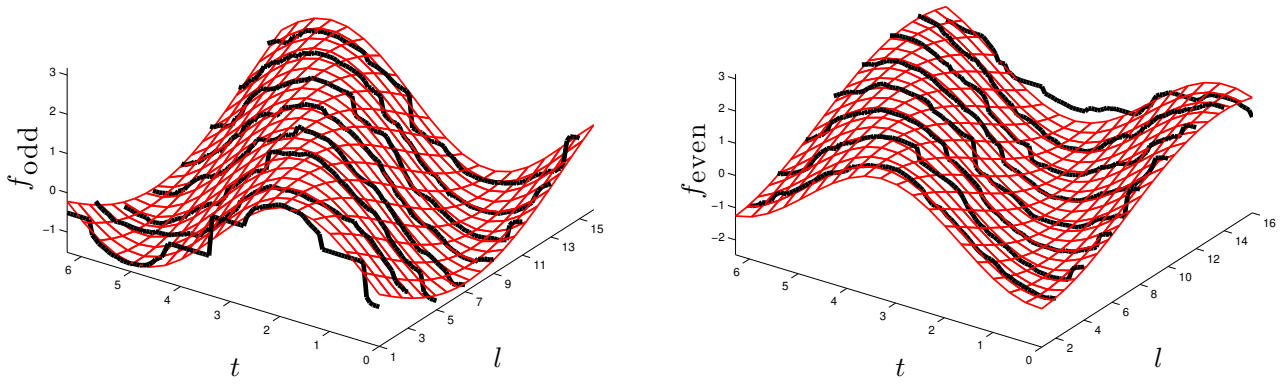


Fig. 8: Waveform for a rolling helix gait. At each time t, the first link of the modsnake is made tangent to the helix with a twist angle of $\phi = t$, and the remaining links are fit to the curve, following the pattern in Fig. 4. For this gait, the joint angle varies sinusoidally with respect to module number as well as time, and thus is described by a pair of traveling waves. As with the rolling arc, the phase shift between the odd and even waves is $\delta = -\pi/2$, which is characteristic of all rolling gaits.

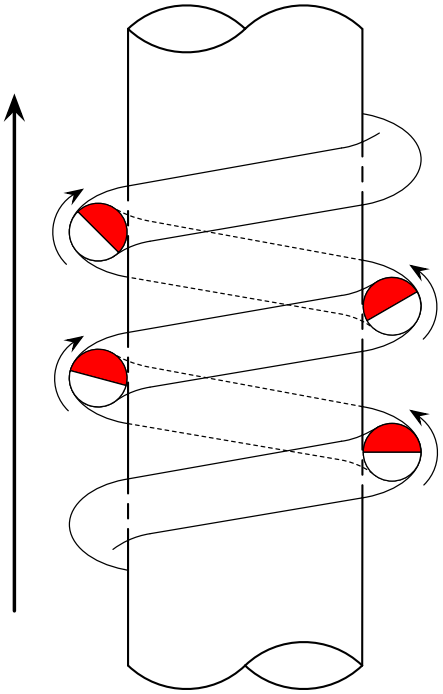


Fig. 9: Rolling up a pole as a helix. In this gait, the snake robot forms a helix around a pole, then twists on its own central axis, so each body segment rolls against the pole with the same net direction of travel.

and is thus not readily extensible to an alternating joint configuration or non-cylindrical geometry. The chain fitting process does not share these requirements, and we can use it to systematically find joint angles for a rolling helix gait without relying on symmetries of the problem.

Generating the rolling helix gait follows much the same process as did generating the rolling arc. At each time t, we place the first link tangent to the helical backbone with twist angle $\phi = t$ and iteratively fit the rest of the links to the curve, as illustrated in Fig. 4. During this fitting, the mechanism takes on a characteristic zig-zag pattern around the backbone curve, which stems from the alternating-joint configuration of the robot. Extracting the odd and even angles gives the plots of g_{odd} and g_{even} in Fig. 8.

The fit angles for the helix gait are noisier than those for the arc, owing to the more complex curvature of the helix increasing the likelihood of the chain fitter becoming stuck in local minima. Even so, a clear pattern emerges, and the joint angles can be seen to vary sinusoidally with respect to both module number and time. This pattern is a traveling wave, and the continuous functions f_{odd} and f_{even} in Fig. 8 are of the form in (1), with parameters $A_{\text{odd}} = A_{\text{even}}$, $\beta_{\text{odd}} = \beta_{\text{even}} = 0$, $\Omega_{\text{odd}} = \Omega_{\text{even}}$, $\omega_{\text{odd}} = \omega_{\text{even}}$ and $\delta = -\frac{\pi}{2}$. These parameters match our previous empirical results [11], validating that chain fitting at various twist angles indeed produces the rolling helix gait used to climb the pole in Fig. 1. Note that the phase shift of $\delta = \pm\frac{\pi}{2}$ is characteristic of rolling gaits, and reflects the fact that the odd and even modules exchange their direction of bending at every quarter-rotation.

C. Rolling in an "S"

Some gaits, even simple ones, cannot be described by the single-wave model, and our new approach reveals appropriate models to use to generate these gaits. For instance, consider a variation on the rolling gait, in which the robot assumes the "S" shape in Fig. 10. No choice of parameters in (1) will produce this motion, but by applying chain fitting and wave extraction to this gait, as in Fig. 11, we can see a clear pattern emerge. The two sets of joint angles are

Fig. 10: Modsnake in its rolling "S" configuration

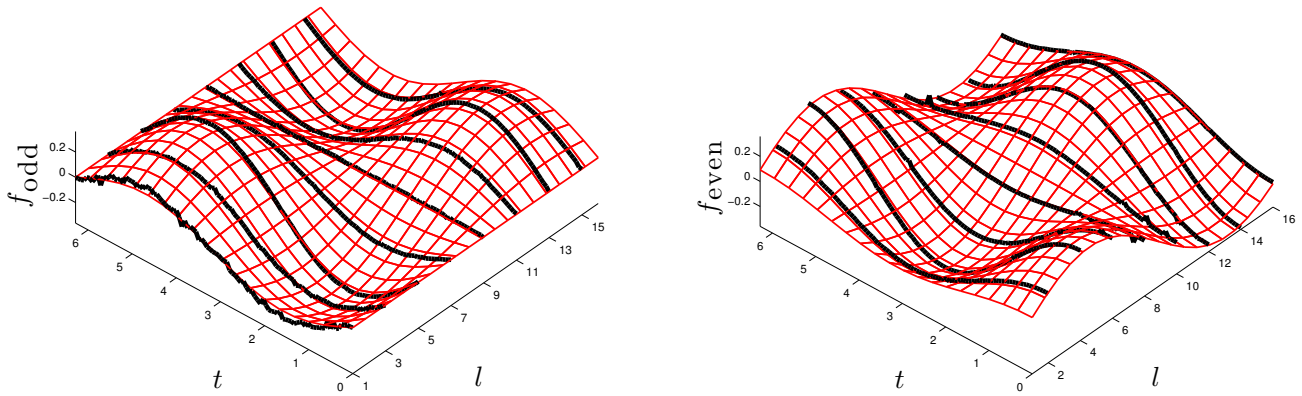


Fig. 11: Wave functions for a rolling "S" gait. The joint angles are described by standing waves, in which the spatial and temporal components are multiplied together, as compared to the additive traveling waves which produce the rolling helix.

described by parameterized functions of the form

$$f_{\text{odd}}(l,t) = A_{\text{odd}} \sin\left(\Omega_{\text{odd}} l + \Delta_{\text{odd}}\right) \sin\left(\omega_{\text{odd}} t\right) \quad (7)$$

$$f_{\text{even}}(l,t) = A_{\text{even}} \sin\left(\Omega_{\text{even}} l + \Delta_{\text{even}}\right) \sin\left(\omega_{\text{even}} t + \delta\right), \quad (8)$$

where the spatial and temporal components are multiplied together, rather than added as in the previous gaits, and Δ and δ represent the spatial and temporal shifts, respectively.

VI. Conclusions

Annealed chain fitting is an effective algorithm for finding the joint angles to fit a long kinematic chain to a backbone curve, and works even when the joints do not allow for omnidirectional bending. Combined with keyframe wave extraction, it provides an attractive means of generating gaits for snake robots. Together, the two algorithms enable both translating a designer's intuition for three-dimensional curves into the space of joint angles, and facilitate the identification of simple parameterized functions that describe those angles.

In this paper, we have explored the use of chain fitting and wave extraction to generate rolling gaits for snake robots that crawl and climb. The waveforms we found match our previous empirical results, confirming the basic principle of our approach. In our future work, we will apply the two algorithms to gaits such as sidewinding and slithering, where the backbone curve itself changes, rather than just the twist angle of the modules. We will be particularly interested in novel gaits which cannot be described by our single-wave model, and are thus beyond the scope of our previous design efforts. We will also look to improving the quality of the chain fitting results with better boundary conditions at the starting end and more nuanced metrics of how well the mechanism fits to the backbone for a given set of joint angles.

Acknowledgements

We would like to thank Matthew Tesch and Marissa Jacovich for their input on this work.

References

[1] S. Hirose, *Biologically Inspired Robots (Snake-like Locomotor and Manipulator)*. Oxford University Press, 1993.

[2] M. Yim, S. Homans, and K. Roufas, "Climbing with Snake-Like Robots," *IFAC Workshop on Mobile Robot Technology*, 2001.

[3] M. Mori and S. Hirose, "Three-dimensional Serpentine Motion and Lateral Rolling by Active Cord Mechanism ACM-R3," in *IROS*, 2002.

[4] C. Wright, A. Johnson, A. Peck, Z. McCord, A. Naaktgeboren, P. Gianfortoni, M. Gonzalez-Rivero, R. Hatton, and H. Choset, "Design of a Modular Snake Robot," in *Proceedings of IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, San Diego, CA, USA, Oct 29 - Nov 2 2007, pp. 2609–2614.

[5] J. Gonzalez-Gomez, H. Zhang, E. Boemo, and J. Zhang, "Locomotion Capabilities of a Modular Robot with Eight Pitch-Yaw-Connecting Modules," in *9th International Conference on Climbing and Walking Robots.*, 2006.

[6] J. Gonzalez-Gomez, H. Zhang, and E. Boemo, "Locomotion Principles of 1D Topology Pitch and Pitch-Yaw-Connecting Modular Robots," in *Bioinspiration and Robotics: Walking and Climbing Robots*. Advanced Robotics Systems International and I-Tech Education and Publishing, 2007.

[7] G. Goldman and D. W. Hong, "Determination of Joint Angles for Fitting a Serpentine Robot to a Helical Backbone Curve," in *International Conference on Ubiquitous Robots and Ambient Intelligence*, November 2007.

[8] ——, "Considerations for Finding the Optimal Design Parameters for a Novel Pole Climbing Robot," in *ASME Mechanisms and Robotics Conference*, August 2008.

[9] M. Sfakiotakis and D. Tsakiris, "Biomimetic Centering for Undulatory Robots," *International Journal of Robotics Research*, November/December 2007.

[10] G. Chirikjian and J. Burdick, "Kinematics of Hyper-redundant Locomotion with Applications to Grasping," in *International Conference on Robotics and Automation*, 1991.

[11] K. Lipkin, M. Tesch, R. Hatton, and H. Choset, "Parameterized and Scripted Gaits for Modular Snake Robots," *Advanced Robotics*, 2009 (accepted).

[12] K. Lipkin, I. Brown, A. Peck, H. Choset, J. Rembisz, P. Gianfortoni, and A. Naaktgeboren, "Differentiable and Piecewise Differentiable Gaits for Snake Robots," in *Proceedings of IEEE/RSJ Intl. Conference on Intelligent Robots and Systems*, San Diego, CA, USA, Oct 29 - Nov 2 2007, pp. 1864–1869.

[13] A. J. Ijspeert, "Central Pattern Generators for Locomotion Control in Animals and Robotics," *Neural Networks*, vol. 21, pp. 642–653, 2008.

[14] J. Burdick, J. Radford, and G. Chirikjian, "A "Sidewinding" Locomotion Gait for Hyper-redundant Robots," *Robotics and Automation*, vol. 3, pp. 101–106, 1993.

[15] S. B. Andersson, "Discrete Approximations to Continuous Curves," in *IEEE International Conference on Robotics and Autmation*, 2006.