

Chlorochlamys Loop-like Locomotion: Combining Crawling and Climbing Robotics*

Konstantinos Karakasiliotis, Leonidas Kagkarakis and Michail G. Lagoudakis

Intelligent Systems Laboratory

Department of Electronic and Computer Engineering

Technical University of Crete

Chania, Crete, Greece

{karakasiliotis, kagkarakis, lagoudakis}@intelligence.tuc.gr

Abstract—Robotic prototypes for search and rescue operations tend to imitate crawling and swimming organisms such as snakes, salamanders, worms, and eels. The Chlorochlamys Chloroleucaria larvae move themselves by a unique form of loop-like body changes stabilized by their subterminal grasping tools; we call this kind of motion loop-like locomotion. By combining undulatory locomotion with loop-like locomotion, robotic prototypes may improve their efficiency and flexibility in moving through unstructured environments, while the climbing gaits may step up their gait repertoire. In our research we study the feasibility of robotic loop-like locomotion and we build robotic prototypes with the above capabilities. We model the Chlorochlamys Chloroleucaria as a multi-segment manipulator with grippers at both ends and we study the motion planning problem for loop-like locomotion under physical and environmental constraints. Extensive experimental studies demonstrate the feasibility and show the effectiveness of the proposed approach. Our robotic prototype is proposed as a testbed for realizing loop-like locomotion in the real world.

Index Terms—Biomimetic Robotic Prototypes, Undulatory Locomotion, Loop-like Locomotion, Motion Planning

I. INTRODUCTION

Our daily need for robust locomotion through unstructured and hostile environments has brought forth the field of biomimetic robotics. The observation and study of biological organisms, which have efficiently solved the problem of locomotion in such environments through the evolutionary process, yields to the imitation and construction of adaptable and robust robotic prototypes.

Robotic prototypes for search and rescue operations tend to imitate crawling and swimming organisms such as snakes, salamanders, worms, and eels [1]–[8]. Experiments with a variety of undulatory robotic prototypes on demanding and laborious substrates (sand, rubble and grass) substantiate the significance of undulatory locomotion as a solution to the locomotion problem in these environments. Even though such prototypes have proved to be more effective than wheeled robots, they fall short of solving the generic locomotion problem during search and rescue operations. For example, granted that an earthquake causes debris of variable elevation, undulatory robots would not be able to operate effectively because of their inability to climb.

Turning back to nature, a small caterpillar, the Chlorochlamys Chloroleucaria larva (Figure 1), moves fluently through the leafy plants and trees as well as climbs up to the earthen pebbles of the plow. Its locomotion ability is a unique form of loopy body changes stabilized by its subterminal grasping tools. This kind of motion is realized by its versatile segmented body while the two subterminal quadruplets of leg-like appendages are able to grasp, stabilizing the body at one or both ends. We call this kind of motion *loop-like locomotion*.



Fig. 1. Chlorochlamys Chloroleucaria larvae moving on leaves.

Drawing inspiration from this dexterous larva, we study the feasibility of robotic loop-like locomotion. We believe that by combining the climbing ability of the Chlorochlamys Chloroleucaria larva with undulatory locomotion may lead to the development of robotic systems able to negotiate unstructured and graded terrain. The combination of these two types of locomotion (undulatory and loop-like) in a robotic prototype in a sense drives biomimetic robotics from *in vitro* to *in vivo* operations.

The remainder of the paper is organized as follows. Section II describes the mathematical equations for undulatory gaits. Section III describes the mechanical model of the *Larvabot*, our robotic prototype that imitates the Chlorochlamys Chloroleucaria larva, and the forward kinematics equations of the mechanism. The proposed algorithm for multi-dimensional motion planning used to generate loop-like gaits under several constraints, is presented in section IV, while generated gaits are demonstrated in section V.

*This work was partially supported by EU Grant MC-IRG-044980.

II. UNDULATORY GAITS GENERATION

A. Snake-like Gaits

Snake-like locomotion mechanisms undulate in parallel to the surface of the substrate. For a snake-like mechanism with N segments and $N - 1$ joints (Figure 2), the motion equation describing the angle $\phi_i(t)$ between consecutive segments with respect to time t is

$$\phi_i(t) = A \sin(2\pi ft + (N - i)\phi_{lag}) + \psi, \quad \text{for } i = 1, \dots, N - 1,$$

where A is the maximum angular deflection of the joints, f the frequency of the traveling wave, ϕ_{lag} the phase lag, and ψ the offset. When the phase lag ϕ_{lag} equals $2\pi/N$, the wave period corresponds to the mechanism's length. The sign of ϕ_{lag} , determines the direction of the traveling wave along the body. The offset ψ and its sign, control the curvature and the direction of the curved paths of the mechanism. Although these parameters can control the direction of the wave, recent studies have shown that the type of interaction with the substrate is in fact responsible for the mechanism's final direction [9].

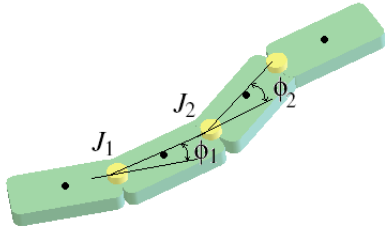


Fig. 2. Undulatory snake-like mechanism.

B. Caterpillar-like Gaits

In contrast to snake-like mechanisms, caterpillar-like locomotion mechanisms riffle transversely to the substrate. For a caterpillar-like mechanism with N segments and $N - 1$ joints (Figure 3), the motion equation is

$$\phi_i(t) = A \sin(2\pi ft + (N - i)\phi_{lag}), \quad \text{for } i = 1, \dots, N - 1,$$

which is almost identical to, and uses the same parameters as, the equation for snake-like mechanisms, excluding the angular offset ψ . These mechanisms do not have the ability to locomote along curved paths, however they have been proved to be very helpful when operating in narrow tubes or passages. They are also very convenient for control, since the mechanism's direction exclusively depends on the sign of ϕ_{lag} and, therefore, it is independent of the type of interaction with the substrate.

III. MODELING LOOP-LIKE LOCOMOTION

A. The *Chlorochlamys Chloroleucaria* Model

The *Chlorochlamys Chloroleucaria* larva is modeled as a serial linkage of N segments ($N - 2$ for the body and 2 for the subterminal tools), interconnected by $N - 1$ consecutive joints

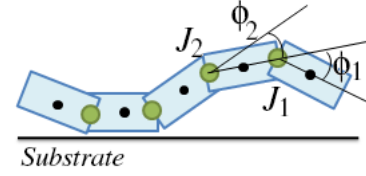


Fig. 3. Undulatory caterpillar-like mechanism.

J_i , $1 \leq i \leq N - 1$, whose angles are denoted by ϕ_i (Figure 4). Each joint has a limited revolutionary ability, therefore $|\phi_i| \leq \phi_{max}$, $1 \leq i \leq N - 1$, where ϕ_{max} is the limit of the joint angles. The length of each segment, characterized by the distance between consecutive joints, is denoted by A_i , $1 \leq i \leq N$.

In order to derive simple loop-like gaits on even surfaces, we view the subterminal tools as footings for standing. In particular, for each of the two ends, the *Stability Region* (denoted by *SR*) is defined as a small region along the x -axis centered at the corresponding endpoint and describes conveniently the function of the subterminal tool in the 2-dimensional vertical space.

The vector $\phi = (\phi_1, \phi_2, \dots, \phi_{N-1})$, is referred to as the configuration of the mechanism and characterizes the pose of its body at any time. Any possible configuration may be valid or invalid; a configuration is characterized as valid if no constraint is violated and invalid otherwise. The constraints relate to system stability and body intersections.

For system stability, when only one end of the body is on the ground, it is required to keep the center of mass of the body within the stability region of that end. All configurations that move the center of mass outside the stability region are automatically characterized invalid as they can cause body flips. Note that any configuration that keeps both ends on the ground is valid as long as the center of mass remains within or between the stability regions of the two ends.

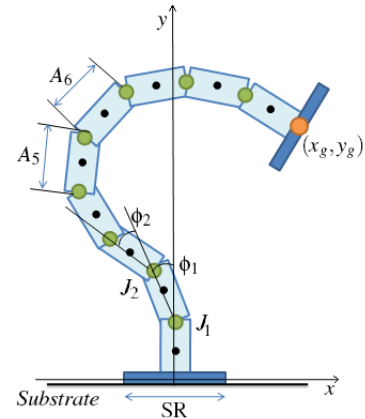


Fig. 4. Loop-like locomotion mechanism model.

The great flexibility of the body may lead to body segments intersecting each other. Given the piece-wise linear nature of the mechanism's body, violation of the body intersection

constraint is determined by checking whether the point, where any two segments of the body intersect, is enclosed between their ends. The coordinates (x, y) of the point of intersection, given the coordinates (x_1, y_1) and (x_2, y_2) of the two ends of the first segment and the coordinates (x_3, y_3) and (x_4, y_4) of the two ends of the second segment, are calculated by

$$x = \frac{\begin{vmatrix} x_1 & y_1 & x_1 - x_2 \\ x_2 & y_2 & x_1 - x_2 \\ x_3 & y_3 & x_3 - x_4 \\ x_4 & y_4 & x_3 - x_4 \end{vmatrix}}{\begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}}, \quad y = \frac{\begin{vmatrix} x_1 & y_1 & y_1 - y_2 \\ x_2 & y_2 & y_1 - y_2 \\ x_3 & y_3 & y_3 - y_4 \\ x_4 & y_4 & y_3 - y_4 \end{vmatrix}}{\begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}}$$

where the vertical bars indicate the determinant of a matrix.

B. Forward Kinematics

For any given configuration ϕ of the body and considering one end of the body as stable, it is imperative to know the position of the other end in the 2-dimensional workspace. This is the point denoted as (x_g, y_g) in Figure 4. This position can be found by the forward kinematic equations of our mechanism:

$$\begin{aligned} (x_g, y_g) &= (f_x(\phi), f_y(\phi)) = \\ &= \left(\sum_{i=1}^{N-1} \sin \left(\sum_{j=1}^i \phi_j \right), \sum_{i=1}^{N-1} \cos \left(\sum_{j=1}^i \phi_j \right) \right), \end{aligned}$$

where N is the number of segments of the mechanism. Obviously, the same kinematic equations can be used to determine the coordinates of any of the inner joints in the 2-dimensional configuration space.

C. Developed Prototype

We have implemented a robotic prototype of the *Chlorochlamys Chloroleucaria* larva, called the *Larvabot* (Figure 5). The *Larvabot* body consists of nine segments, including the subterminal grasping and standing tools, and eight joints. The active tools at each end of the body consist of three leg-like appendages which can extend and retract as needed according to the current operating mode. This flexibility of the subterminal tools is essential, as we require that our robotic prototype is able to operate in all three modes of locomotion (snake-like, caterpillar-like, loop-like). While the current implementation of the subterminal tools suffices for demonstrating a proof of concept, we believe that extensive experimentation and better design will be necessary for endowing the *Larvabot* with powerful climbing capabilities.

The entire hardware of the *Larvabot* is part of a BIOLOID robot kit¹ which includes several actuators (dynamixels), a programmable controller, and a large variety of mounting brackets. The actuators are two-way communication servos driven by a serial network (TTL) and provide feedback

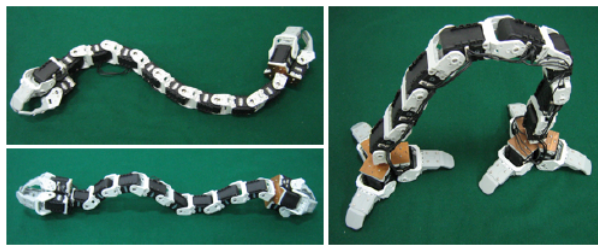


Fig. 5. The *Larvabot* operating in snake-like mode (upper left), in caterpillar-like mode (bottom left) and in loop-like mode (right).

including shaft position, temperature, as well as input and output voltage. When operating at 10V, the servo motors hold torque up to 16.5 kg-cm and a maximum speed of 0.196 sec/60°. The programmable controller is an Atmel ATmega128 which can be programmed using the C language or the higher-level software provided with the kit.

IV. MOTION PLANNING FOR LOOP-LIKE LOCOMOTION

Given that the snake-like and caterpillar-like undulatory gaits have been studied in detail, we have been able to incorporate them fairly easily on the *Larvabot*. However, to implement and incorporate loop-like locomotion, it was necessary to carry out a detailed study of this kind of motion using simulation. The purpose of the research described in this paper is to derive the necessary control laws for achieving loop-like locomotion.

Our goal in this section is to develop planning algorithms for finding appropriate paths in the configuration space of the robot, which yield the desired motion when followed over a period of time. The configuration space of our robot is the $(N-1)$ -dimensional space of angles between the N joints, $\Phi = \{(\phi_1, \phi_2, \dots, \phi_{N-1}) : -\pi \leq \phi_i \leq \pi \text{ for } i = 1, 2, \dots, N-1\}$. Note that Φ contains all possible combinations of angles, including all valid and invalid robot configurations.

The general motion planning problem is to find a path that lies in the valid part of the configuration space and connects an initial configuration to a target configuration. Two observations should be made at this point. First, in our motion planning problem, the target configuration is largely unknown as it is typically given in terms of the position (x_g, y_g) of the moving endpoint of the robot in the physical workspace (Figure 4). The family of configurations that drive the endpoint to the desired position in the workspace cannot be easily derived, as no inverse kinematic model of the robot is available (the problem is similar to inverse kinematics of a highly-redundant manipulator). Second, the high-dimensional nature of the configuration space prevents us from using complete and systematic methods for global motion planning. This is due to the phenomenon known as *curse of dimensionality* (the search space grows exponentially with the number of dimensions). Therefore, our focus will be on heuristic methods which deliver feasible, yet not necessarily optimal, motion paths.

¹Trademark of Robotis, South Korea.

We first study the simplest case of loop-like locomotion, namely locomotion on even terrain with no elevation or obstacles. In this case, the desired motion/gait can be achieved by repeated executions of a *loop step*. During such a step, the Larvabot begins in the upright position with one endpoint on the ground and the body oriented vertically straight up (cf. Figure 1–left) and operates in three phases: (1) it moves the free endpoint to some place forward on the ground, (2) it switches the roles of the two ends by transferring the center of mass from one end to the other, and (3) it moves the new free endpoint to the upright position. A straightforward way to realize this three-phase loop step is to find a path that moves the free end from the upright position to the ground and a path that mirrors the resulting configuration (with both ends on the ground) with respect to the middle of the body. The third phase can be achieved simply by executing the motion path of the first phase in reverse order over time and over joints, given that the third phase begins with the mirrored configuration at the end of the first phase. The following subsections describe in detail how these three phases are implemented.

A. Phase 1 and 3: Gradient Following Approach

Let (x_g, y_g) be the goal position of the endpoint in the workspace. For positions on the ground it will be $y_g = 0$. Given that the objective is to drive the robot to a configuration that brings the endpoint to the goal position, it makes sense to apply a gradient following method for minimizing the following objective function over the parameters $\Phi = (\phi_1, \phi_2, \dots, \phi_{N-1})$:

$$\begin{aligned} F(x_g, y_g, \Phi) &= (f_x(\Phi) - x_g)^2 + (f_y(\Phi) - y_g)^2 = \\ &= \left(\sum_{i=1}^{N-1} \sin \left(\sum_{j=1}^i \phi_j \right) - x_g \right)^2 + \left(\sum_{i=1}^{N-1} \cos \left(\sum_{j=1}^i \phi_j \right) - y_g \right)^2. \end{aligned}$$

This quadratic objective function is globally minimized when the position of the endpoint in the workspace reaches (x_g, y_g) . The gradient of F with respect to each adjustable parameter ϕ_i is given by

$$\begin{aligned} \frac{\partial F}{\partial \phi_i}(x_g, y_g, \Phi) &= 2 \left(\sum_{i=1}^{N-1} \sin \left(\sum_{j=1}^i \phi_j \right) - x_g \right) \cos \left(\sum_{j=1}^i \phi_j \right) \phi_i \\ &\quad - 2 \left(\sum_{i=1}^{N-1} \cos \left(\sum_{j=1}^i \phi_j \right) - y_g \right) \sin \left(\sum_{j=1}^i \phi_j \right) \phi_i \\ &= 2(f_x(\Phi) - x_g) \cos \left(\sum_{j=1}^i \phi_j \right) \phi_i - 2(f_y(\Phi) - y_g) \sin \left(\sum_{j=1}^i \phi_j \right) \phi_i \end{aligned}$$

Therefore, the gradient-following rule for minimizing F moves the current configuration $\Phi(t)$ at each time step t along the gradient at that point using

$$\phi_i(t+1) = \phi_i(t) - \alpha_i \frac{\partial F}{\partial \phi_i}(x_g, y_g, \Phi(t)),$$

where α_i is a small positive number denoting the step size in each dimension i . The rule terminates when $F(x_g, y_g, \Phi(t))$ becomes less than ϵ (a small positive constant).

If there are no local minima between the current and the target configurations in the configuration space due to constraint violations, the above gradient-following rule will eventually drive the current configuration to the target configuration. The stability constraint yields multiple local minima for virtually all target configurations needed for implementing the first phase of our loop step. To overcome this difficulty, we incrementally apply small perturbations in the opposite direction to the joints closer to the stable end as a way of bringing the center of mass back to the stability region and exiting from such minima. This simple trick suffices for deriving motion paths appropriate for the first phase of our loop gait.

B. Phase 1 and 3: Randomized Heuristic Search Approach

An alternative approach to finding the desired path in the configuration space is to use search techniques [10]. In this case, we form a search tree starting with the current configuration as the starting state (root node of the tree) and we expand the tree until the goal state with the target configuration is reached. The path connecting the root node to the leaf node where the target was found forms the path for the desired motion.

The most delicate issue in this approach is the definition of the successor function, that is, the generation of the configurations directly reachable from the current configuration, which will become the children of the current node in the search tree. Given that the configuration space is high-dimensional and continuous, it would be impossible to generate all possible successors. Even a coarse discretization of the configuration space would result in an exponential number of successors with respect to the number of dimensions. Expansion of the search tree beyond the first few levels would be impossible with such a huge branching factor.

The solution we adopted was to restrict the number of successors to a fixed number K ; these K successors are drawn uniformly in random from a hypersphere of radius R_i in each dimension i , $1 \leq i \leq N-1$, centered at the current configuration. Successors corresponding to invalid configurations are discarded and the corresponding branch of the search tree is pruned. This choice for the successor function strikes a balance between sampling sufficiently the space of successors and keeping the branching factor of the search tree low. It nevertheless compromises the systematic coverage of the configuration space and therefore any claims for optimality of the solution, a cost which can certainly be paid for the sake of feasibility.

All generated nodes are scored using a simple heuristic function which computes the Euclidean distance between the current and the target position of the free endpoint in the work space of the robot. In particular, a node corresponding to configuration Φ is scored by

$$h(x_g, y_g, \Phi) = \sqrt{(f_x(\Phi) - x_g)^2 + (f_y(\Phi) - y_g)^2},$$

where (x_g, y_g) is the goal position of the free endpoint in the workspace. All pending, unexpanded nodes are kept in

a priority queue according to their heuristic value. During search, the node with the minimum heuristic value in the queue is expanded. Therefore, the search procedure gives priority to configurations that seem to be closer to the target configuration. As before, the search procedure terminates when $h(x_g, y_g, \phi)$ becomes less than ϵ (a small positive constant).

Once again, this is not a complete algorithm and might get stuck in local minima, however it suffices for realizing the first phase of our loop step. In contrast to the gradient-following approach, the randomized heuristic search approach offers two advantages: (1) it can easily incorporate any metric other than the Euclidean distance in the workspace to optimize during the search, such as Euclidean distance between configurations or locations of the center of mass, and (2) it can be extended, albeit with significant additional work, to a systematic method that overcomes local minima and yields paths from any initial valid configuration to any other valid configuration.

C. Phase 2: Configuration Mirroring

The second phase of our loop step is realized by a simple technique, namely mirroring of the current configuration over the middle of the body of the robot. During this phase both ends are stably placed on the ground, therefore the center of mass can lie anywhere between the two endpoints without violating the stability constraint. In effect, after mirroring, the center of mass moves from the stability region of the previously stable end to the stability region of the other end which will become the stable one in the next loop step.

Given a configuration

$$\phi = (\phi_1, \phi_2, \phi_3, \dots, \phi_{N-2}, \phi_{N-1})$$

where both ends lie on the ground and the center of mass is within the stability region of the first segment, the mirrored configuration is

$$\hat{\phi} = (\theta, \phi_{N-1}, \phi_{N-2}, \dots, \phi_4, \phi_3, \phi_2)$$

where

$$\theta = \begin{cases} +\pi - \sum_{i=1}^{N-1} \phi_i, & \text{if } \sum_{i=1}^{N-1} \phi_i \geq 0 \\ -\pi - \sum_{i=1}^{N-1} \phi_i, & \text{if } \sum_{i=1}^{N-1} \phi_i < 0 \end{cases}$$

It is easy to see that indeed $\hat{\phi}$ is the mirrored configuration, except perhaps for the angle θ of the first joint. To understand this, notice that the first angle in the mirrored configuration must assume a value equal to the angle θ of the last segment with respect to the vertical negative axis at the moving endpoint. This particular angle is not part of configuration ϕ , however it can be easily computed. Since each angle ϕ_i is relative to the previous segment at joint i , the sum of all ϕ_i 's essentially represents the total rotation of the last segment with respect to the vertical positive axis at the stable endpoint, therefore θ is conveniently computed as the complimentary angle of the total rotation.

A smooth transition from the initial configuration ϕ to the mirrored configuration $\hat{\phi}$ that keeps both endpoints stable

can be achieved by simply moving along the straight line connecting the two configurations in the configuration space. If M is the number of steps allowed along this path, then the configuration is updated as follows:

$$\phi_i(t+1) = \phi_i(t) + \frac{\hat{\phi}_i(M) - \phi_i(0)}{M},$$

where $\phi(0)$ is the initial configuration and $\hat{\phi}(M)$ is the mirrored configuration achieved after M steps. All configurations along this path are guaranteed to be valid as the center of mass moves between the two ends that are stably placed on the ground.

V. EXPERIMENTAL RESULTS

In this section, we focus on simulation results of loop-like locomotion that demonstrate the feasibility and applicability of our approach.

Figure 6 shows snapshots of a complete loop step using the gradient following approach with $N = 8$, $A_i = 5$, $\epsilon = 1$, and $\alpha_i = 0.0003N$. Notice that the step size has a smaller value at the joints closer to the stable end and a larger value at the joints closer to the moving end. This choice deemed necessary in order to make gradient steps small enough to eliminate jerkiness of motion. As demonstrated in the figure, during the first phase, the robot moves from the upright position at $(0, 40)$ to the target position $(x_g, y_g) = (15, 0)$, then during the second phase it mirrors the configuration thus transferring the center of mass from the $(0, 0)$ endpoint to the $(15, 0)$ endpoint, and finally during the third phase it moves the $(0, 0)$ endpoint to the upright position at $(15, 40)$. Even though it is not obvious in the figure, the motion is somewhat jerky mostly because of the perturbations needed for exiting local minima. Another weakness of this approach is the great variability in velocity; the value of the gradient causes large steps away from the target and tiny steps when approaching the target.

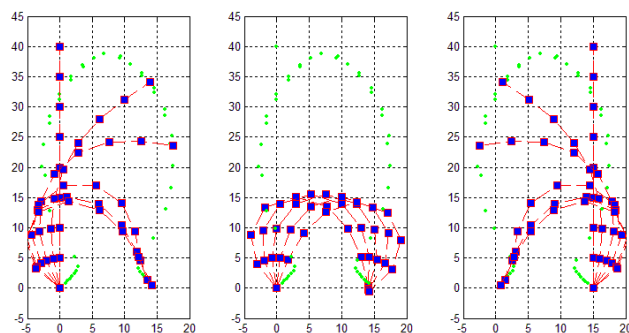


Fig. 6. Snapshots of one loop step on even terrain using the gradient following approach. The blue squares indicate the joints while the green path delineates the trajectories of the endpoints.

Figure 7 shows snapshots of a complete loop step using the randomized heuristic search approach with $N = 8$, $A_i = 5$, $K = 200$, $\epsilon = 0.1$, and $R_i = 20^\circ / (N - 1 - i)$. Notice that the hypersphere has a smaller radius at the joints closer to the

stable end and a larger radius at the joints closer to the moving end. This choice effectively removes jerkiness of motion as it allows for greater flexibility in search only where needed (closer to the free endpoint). As demonstrated in the figure, the robot moves smoothly from the upright position to the target position $(x_g, y_g) = (15, 0)$ and up again, as before. The main difference with the gradient-following approach is the smoothness of motion. In this case, there is no great variability in velocity during the motion, nor is there backtracking due to perturbations. As a result, the completion time of the loop step is about 8 times smaller when using the randomized search approach.

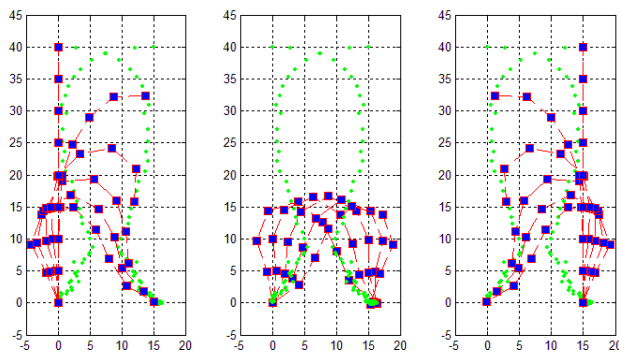


Fig. 7. Snapshots of one loop step on even terrain using the randomized heuristic search approach. The blue squares indicate the joints while the green path delineates the trajectories of the endpoints.

The associated video clip shows the simulation of three complete loop steps with both approaches. At each step the body moves 15 units to the right. The yellow square shown on the video indicates the location of the center of mass at any time. It should be noted that the simulation for the gradient-following approach has been sped up by a factor of 8 to make the two animations comparable. In addition, it shows snake-like, caterpillar-like, and loop-like locomotion on a simplified version of the Larvabot that resembles the model in Figure 4.

VI. CURRENT AND FUTURE WORK

We are currently working on integrating real-time motion planning of the loop-like gait on the Larvabot. The main challenge in this work is the limited computational power of the controller which calls for careful optimization of the proposed algorithms. We are also working on extending the randomized heuristic search approach to a complete and systematic search method for motion planning. The main issue here is the definition of an appropriate cost function and the avoidance of repeated states, that is, the ability to avoid re-visiting parts of the configuration space that have already been explored. This goal could be possibly accomplished using Rapidly-Exploring Random Trees (RRTs) [11], the technique we are currently investigating. Finally, we are working on a motion planning algorithm for loop-like locomotion on uneven (graded) terrain. The three-phase step proposed in this

paper can be adapted appropriately in this case. Phases 1 and 3 remain largely unchanged as the proposed planning algorithms can accommodate placement of the moving endpoint to any elevated target position ($y_g \neq 0$). The tricky part relies in Phase 2 where simple configuration mirroring is not sufficient any more to transfer the center of mass from one end to the other.

In the future, we plan to enhance the Larvabot with sensing capabilities, such as bump sensors at the two grasping ends and a small CCD camera inside one of the two grasping ends. In addition, we plan to add revolutionary joints right before the grasping tools; these joints will enable the Larvabot to rotate the body while one end is stably placed on the ground, thereby allowing for locomotion towards any direction on the plane. Finally, we are considering replacing or enhancing the current grasping appendages with stronger grippers (e.g. pneumatic) which will enhance the Larvabot with larger stability regions.

VII. CONCLUSION

Inspired by the abilities of the *Chlorochlamys Chloroleucaria* larva, we proposed motion planning algorithms for loop-like locomotion and we designed the Larvabot, a robotic prototype which combines snake-like, caterpillar-like, and loop-like locomotion on the same body. We believe that robots with such combined crawling and climbing motion abilities can play a crucial role as assistants in search-and-rescue missions.

REFERENCES

- [1] S. Hirose, *Biologically Inspired Robots: Snake-like locomotors and manipulators*. Oxford University Press, 1993.
- [2] S. Hirose and E. F. Fukushima, "Snakes and strings: New robotic components for rescue operations." *International Journal of Robotic Research*, vol. 23, no. 4-5, pp. 341-349, 2004.
- [3] M. Saito, M. Fukaya, and T. Iwasaki, "Serpentine locomotion with robotic snakes," *IEEE Control Systems Magazine*, vol. 22, pp. 64-81, 2002.
- [4] J. Cortés, S. Martínez, J. P. Ostrowski, and K. A. McIsaac, "Optimal gaits for dynamic robotic locomotion," *International Journal of Robotic Research*, vol. 20, no. 9, pp. 707-728, 2001.
- [5] J. Ostrowski and J. Burdick, "The geometric mechanics of undulatory robotic locomotion," *International Journal of Robotic Research*, vol. 17, no. 7, pp. 683-701, 1998.
- [6] A. Ijspeert, A. Crespi, and J. Cabelguen, "Simulation and robotics studies of salamander locomotion. Applying neurobiological principles to the control of locomotion in robots," *Neuroinformatics*, vol. 3, no. 3, pp. 171-196, 2005.
- [7] M. Sfakiotakis and D. P. Tsakiris, "Neuromuscular control of reactive behaviors for undulatory robots," *Neurocomput.*, vol. 70, no. 10-12, pp. 1907-1913, 2007.
- [8] J. Yu, L. Liu, and L. Wang, "Dynamics and control of turning maneuver for biomimetic robotic fish," in *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, October 2006, pp. 5400-5405.
- [9] M. Sfakiotakis, D. P. Tsakiris, and K. Karakasiliotis, "Polychaete-like pedundulatory robotic locomotion," in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007, pp. 269-274.
- [10] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003.
- [11] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *International Journal of Robotic Research*, vol. 20, no. 5, pp. 378-400, 2001.