Planning with the STAR(s)

Konstantinos Karydis, David Zarrouk, Ioannis Poulakakis, Ronald S. Fearing and Herbert G. Tanner

***Abstract*— We present our findings on the first application of motion planning methodologies to the recently introduced Sprawl Tuned Autonomous Robot (STAR). The reported results provide a first glimpse on the capabilities of this novel, 3D-printed robot in performing autonomously non-trivial motion planning tasks in environments populated with obstacles. We employ methods from sampling-based motion planning under nonholonomic constraints, and implement in open loop the generated path on the physical robot for various environments of increasing complexity.**

I. Introduction

Advances in materials and manufacturing processes have made possible the introduction of a large number of miniature legged robots. The subject of our work, the 3D-printed STAR robot [1], constitutes one such example. Other examples include the cockroach-inspired hexapod [2] that uses two piezoelectric ceramic actuators to drive its legs, the Mini-Whegs robot series [3], [4], utilizing a three-spoke rimless wheel ("wheg"), and the i-Sprawl robot [5], manufactured via Shape Deposition Manufacturing (SDM) [6]. Minimally actuated walkers have been built through the Smart Composite Microstructure (SCM) fabrication technique [7]; examples include the crawlers DASH [8], DynaRoACH [9], VelociRoACH [10], and OctoRoACH [11].

The increasing interest in miniature legged robots primarily lies in their promising mobility capabilities. Indeed, they can traverse uneven terrain where wheeled robots may fail, and enter confined environments such as caves, crevices, and collapsed buildings. Compared to larger legged robots, they also constitute a cost-effective means to perform exploration and reconnaissance missions since their low cost and production time allow for rapid manufacturing and deployment in large numbers. As a result, failure of some platforms may be acceptable, as long as a required task is completed. Given these opportunities and potential, research has started to focus on problems related to motion planning and autonomous navigation for miniature crawlers [12], and the work presented in this paper is along these lines.

The Sprawl Tuned Autonomous Robot (STAR), which we use in this study, is a light-weight, high-mobility 3D-printed robot, designed for low cost and rapid mass production. It is constructed out of a kit of parts, thus making it easy to assemble and repair; full assembly of its mechanical parts requires about 30 min. The robot is equipped with three-spoke rimless wheel legs and a mechanism that adjusts the posture of the robot by changing the sprawl angle of its legs from nearly flat posture to vertically oriented legs. Due to its reconfigurable sprawl angle, STAR possesses high-mobility capabilities combining the benefits of wheeled and legged locomotion. In particular, large sprawl angles were found to be efficient on uneven terrain, whereas the low sprawl posture is better suited for traveling over smooth surfaces, performing similar to wheeled vehicles [1], [13]. These properties allow for actuation with low energy requirements [14], an asset in the context of autonomous motion planning.

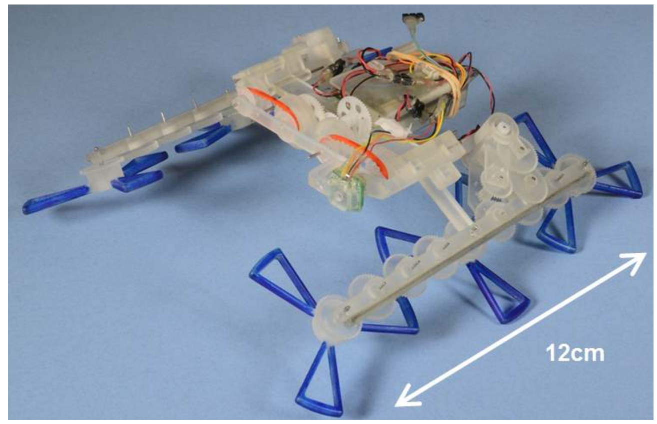


Fig. 1. The 3D-printed Sprawl Tuned Autonomous Robot (STAR). Its main body has 12 cm length and 11 cm width, and weights about 70 g. Its height is adjusted by changing the sprawl angle which is actuated independently of the legs. The robot can run stably up to a maximum speed of 5 m/s.

A. Motion Planning on Crawling Robots

To plan the motion of a legged robot, we need a model that adequately captures mathematically and reproduces this motion in simulation, and a few such models already exist. The Spring Loaded Inverted Pendulum (SLIP) [15], [16] model has demonstrated its ability to capture the motion of animals and robots of various morphologies in the sagittal plane. Specifically to miniature crawling robots, various bio-inspired, horizontal-plane modeling approaches have been proposed. The Lateral Leg Spring (LLS) model [17]–[19] is a conservative mechanical system that explains lateral stabilization [20], and has been also used in deriving turning strategies [21], [22] for hexapedal runners. To account for the sliding effects of the leg-ground interaction, the Sliding Spring Leg (SSL) model [23] has been proposed. Moreover,

This work is supported in part by NSF under grants CMMI-1130372 and CNS-1035577, and by ARL MAST CTA # W911NF-08-2-0004.

Konstantinos Karydis, Ioannis Poulakakis and Herbert Tanner are with the Department of Mechanical Engineering, University of Delaware. Email: {kkaryd, poulakas, btanner}@udel.edu

David Zarrouk is with the Department of Mechanical Engineering, Ben Gurion University, Beer Sheva, Israel. Email: zadavid@post.bug.ac.il

Ronald Fearing is with the Department of Electrical Engineering and Computer Science, University of California, Berkeley. Email: ronf@eecs.berkeley.edu

the in-plane behavior of miniature robots while crawling at low speeds is captured on average by a kinematic template model [24], which allows for a direct mapping of high-level motion planning specifications to robot parameters.

The design of STAR renders compliance practically negligible. Also, it has been found that STAR moves like a wheeled vehicle at low sprawl postures [13]. For these reasons, and due to STAR's ability to combine wheeled and legged locomotion, its behavior when operating at low sprawl angles may be adequately approximated at a first-order scale by a car-like, planar model.

B. Overview

In this work, we restrict our analysis to the horizontal plane, keeping the sprawl angle fixed at 30^o. Due to its differential-drive steering method, the robot is modeled as a *Dubins* vehicle [25]. For this model, existing planning methodologies [26], [27] can be brought to bear to tackle non-trivial planning problems. In particular, we adopt an approach based on *motion primitives*; in this method, a small number of basic mobility behaviors exhibited by the robot are concatenated together to produce more complex motion.

The motion primitives for our case are three: (i) *go straight* (S), (ii) *hard turn left* (L), and (iii) *hard turn right* (R). The robot undergoes an initial calibration stage, where position and orientation data is collected via motion capture in order to experimentally identify the path profiles for the primitives we consider. Then, we use the data-driven primitives thus constructed in conjunction with an RRT planner [28] to plan the motion of STAR in a series of cluttered environments.

The outcome of the planner is used in open loop to drive the physical robot from an initial configuration to a desired one. Experiments demonstrate that the uncompensated accumulated error makes actual paths deviate substantially from the planned path, and thus the robot comes in contact with the workspace boundary and obstacles. In most of these cases, however, the robot was still able to progress toward the goal.

The paper contributes to the area of small legged robot navigation and planning, and demonstrates the first implementation of primitives-based motion planning strategies to the novel STAR robot. Its capabilities make it a promising example of this class of robots, tasked to perform exploration duties in confined and unstructured environments. The first step toward achieving these aims involves testings on flat ground in confined spaces, and is treated in this work.

C. Organization

The rest of the paper is organized as follows. Section II presents the STAR robot. Section III focuses on the implementation of a sampling-based planner using motion primitives, and Section IV presents our experimental results on planning with the STAR. Section V concludes the paper.

II. The STAR Robot

The STAR robot, shown in Fig. 1, is a bio-inspired hexapod which incorporates a three-spoke rimless wheel leg design, and a mechanism to adjust its sprawl posture. The main body is 12 cm-long and 11 cm-wide, and its weight is around 70 g. By changing its sprawl angle, the robot motion varies from purely sagittal to mainly lateral, and in the latter the uncontrolled vertical dynamics component is considerably decreased; this component poses some challenges related to stabilization of miniature crawling robots. This way, STAR is able to achieve high-speed horizontal motion, and high maneuverability with low control actuation, thus extending battery life.

A. Design and Manufacturing

The robot has a main body which houses the controller, the battery, and the sprawl-adjusting mechanism. In all six legs, each spoke has a 60^o phase offset between neighboring spokes in order to ensure that the robot has sufficient ground contact at all times during locomotion. The three-spoke leg drives are attached to each side of the robot through a rotational pin joint controlled by the sprawl mechanism. The relative angle between the legs and the main body, as presented in Fig. 2, forms the sprawl angle. The 0^o value is defined at the configuration in which the legs are coplanar with the ground. By convention, the sprawl angle takes positive values when the legs move downward (see Fig. 2). Note that the sprawl angles of both sides are actuated simultaneously through a single motor and a geared mechanism that ensures identical sprawl on both sides.

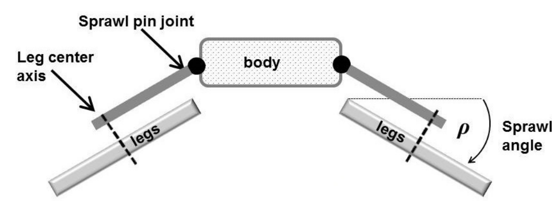


Fig. 2. Front view of the robot. By convention, downward leg rotation produces a positive change in the sprawl angle.

The sprawl angle varies in the interval of $[80^o, -90^o]$; typical sprawl posture configurations are shown in Fig. 3. The sprawling capability allows the robot to overcome obstacles by climbing over, or crawling beneath them. It also allows STAR to continue running even when flipped upside down [13].

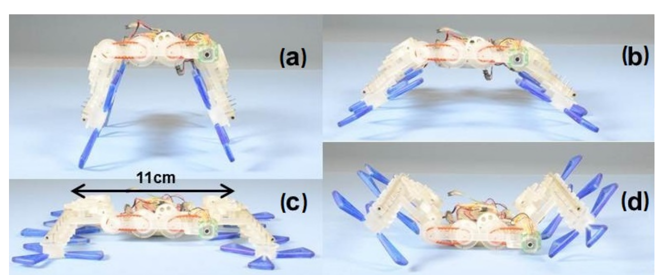


Fig. 3. Different sprawl postures. (a) Positive sprawl angle. (b) Small positive sprawl angle. (c) Zero sprawl angle. (d) Negative sprawl angle.

With the robot being able to develop high travel speed, it is imperative to strengthen its body against collisions. Two 12 cm-long, 2 mm in diameter carbon fiber reinforced rods are embedded along the long side of the leg drive structure (see Fig. 4). During preliminary experimental testing, the robot sustained multiple collisions at speed above 3 m/s, but the leg drive structures did not fail. However, damage did occur on certain occasions to the front legs which had to be replaced. Nonetheless, the rimless wheel leg design allows us to postulate that the robot may be able to still propel itself even when some of the spokes are severed.

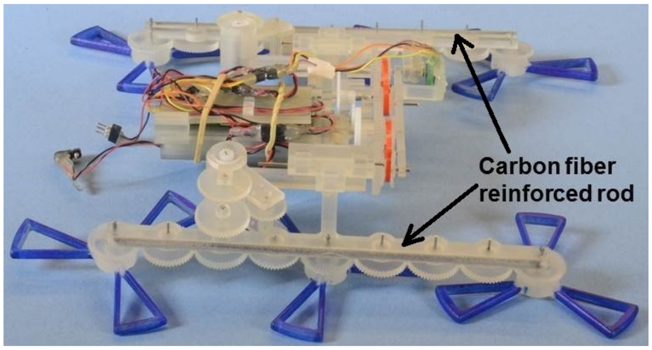


Fig. 4. Side view of STAR. A carbon fiber reinforced rod is used at each side to strengthen the robot's leg drives.

STAR is designed for rapid manufacturing; all of its body parts and legs, and most of its gears, are 3D-printed. The robot is designed for easy assembly and part replacement. The parts are attached together through press fitted pins, and the complete assembly of its mechanical parts requires about 30 min.

B. Actuation

STAR has three motors (Didel MK07-3.3) to drive its legs and sprawl-adjusting mechanism. A spur gear transmission of 1:16 or 1:48 drives the legs depending on the desired speed, and a 1:576 worm gear transmission changes the sprawl angle. With a 300 mA-hr LiPo, 4 V battery, the benchmark time duration for running STAR continuously at full capacity exceeds 30 min. The robot is currently able to carry an extra weight of 100 g without any performance reduction at roughly 1 m/s. Accessories that can be mounted at this weight range include miniature sensors such as a mobile phone camera or a second battery for prolonged life span.

C. Control Architecture and Localization

The robot uses an ImageProc 2.2 controller board [11] that drives its motors, and permits wireless communications to a remote host. The control inputs to the motors driving its legs are motor gains, K_L, and K_R, controlling the leg velocities of the left and the right side, respectively. The two motors are controlled independently, and, as a result, the robot employs a differential-drive steering method for navigation.

Currently, a motion capture system is used for localization by providing ground truth. Ongoing work involves the application of off-the-shelf controller solutions (e.g. Arduino boards), and the addition of range sensors, IMU, and compass for investigating the capabilities of the robot with on-board sensing only.

III. Planning the Motion of STAR

With the sprawl angle fixed at 30º, the robot moves like a wheeled vehicle.[1]

A. A Model for STAR

Due to its differential-drive steering method, optimal paths for the robot moving in the horizontal plane are a combination of *Dubins* curves [25]. For our case, these curves correspond to the three primitives considered; the particular robot motor gains implementing them are included in Table I.

TABLE I
IMPLEMENTED STAR ACTIONS

Type	Description	Motor Gains ($[K_L, K_R]$)
S	Go Straight	$[100, 100]$
L	Hard Turn Left	$[0, 100]$
R	Hard Turn Right	$[100, 0]$

The path profiles of the STAR motion primitives shown in Table I are found experimentally through an initial calibration process. We collect open-loop planar position and orientation measurement data from a total of 30 paths for each primitive with a motion capture system at a rate of 100 Hz. With respect to Fig. 5, the measured states are the planar position of the geometric center of the robot (x_G, y_G), and the heading θ. The initial configuration of the robot for all trials is $(x_G, y_G) = (0, 0)$ cm, and $\theta = 0$º. Table II summarizes the initial pose error statistics.

TABLE II
ERROR IN INITIAL POSE (CALIBRATION STAGE)

Type	Mean [cm, cm, deg]	Standard Deviation [cm, cm, deg]
S	(1.58, 2.27, 0.39)	(1.18, 2.08, 1.25)
L	(1.71, 2.32, −1.68)	(1.47, 1.97, 2.01)
R	(1.30, 2.41, 1.12)	(0.85, 1.02, 1.83)

All trials are conducted on a rubber floor mat surface for a fixed duration of 2 sec. The inherently uncertain interaction between the legs of the robot and the ground makes longer paths impractical, as the variance associated with expected position and orientation measurements becomes unacceptably large over longer time intervals. Capturing the stochasticity induced by the leg-ground contact is outside the focus of this paper; related work is reported in [29].

Figure 6 contains the experimental paths gathered during the calibration phase. The black thick curves correspond to the experimental average for each primitive, and last 2 sec. In the sequel, we use as input to the motion planner (see next section) only the first half (1 sec duration) of these curves; doing so reduces the associated variance in our primitives, and allows for navigation in confined environments.

[1] Solving planning problems for the full spectrum of STAR motion capabilities is part of future work.

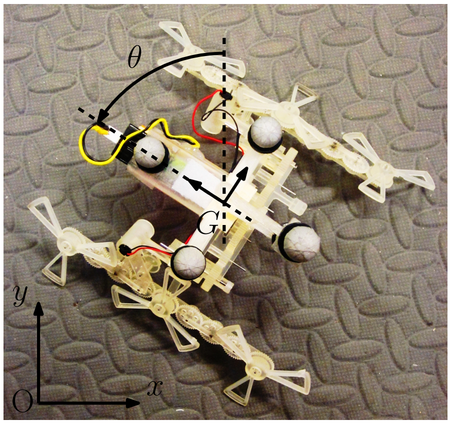


Fig. 5. The state of the robot consists of $(x_G, y_G) \in \mathbb{R}^2$, the position of the geometric center of the model, G, with respect to some inertial coordinate frame, O, and $\theta \in \mathbb{S}^1$, the angle formed with respect to the longitudinal body-fixed axis and the y-inertial axis.

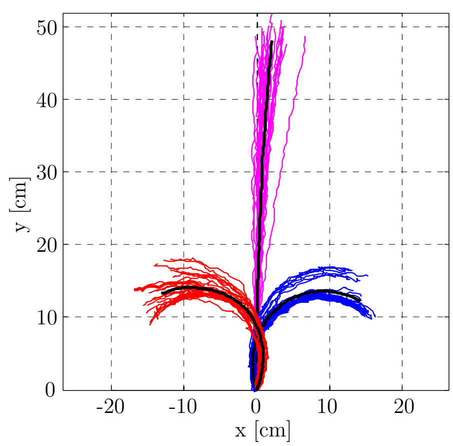


Fig. 6. Experimental data for the STAR-generated Dubin's curves produced by the motor gains contained in Table I. Individual curves depict the evolution of the geometric center of the robot, while the experimental average out of a total of 30 paths for each case is shown in black thick curves. Curves in red, blue, and magenta correspond to the *Hard Turn Left*, *Hard Turn Right*, and *Go Straight* primitives, respectively. Notice the increasing variance as the running time elapses.

B. Sampling-Based Motion Planning

In this study we consider a single robot exploring a static environment; its task is to move from an initial to a target configuration. We implement an approach based on *rapidly exploring random trees* [28] (RRTs), which is able to handle the motion constraints of the robot when navigating in spaces populated with obstacles.[2] Extensions leading to optimal plans [31] are in principle applicable, but we choose the original RRT planner mainly due to its popularity, proven efficacy in experiments, and availability of off-the-shelf software implementing the basic algorithm.

The planner is implemented in three illustrative scenarios depicted in Fig. 7. The complexity of the problem increases from case to case with the addition of obstacles. The initial configuration remains the same in all cases, chosen at $q_0 = (x_0,\ y_0,\ \theta_0) = (110,\ 85,\ 0)$ [cm, cm, deg]. Similarly, the desired configuration is also kept constant, at $q_d = (x_d,\ y_d,\ \theta_d) = (40,\ 80,\ -90)$ [cm, cm, deg]. Due to motion constraints, and the discrete nature of assumed robot behaviors, the planned path ending exactly at q_d is very unlikely; therefore, we consider the target reached when the path ends within a radius of 10 cm around the target position, with final orientation in the range of $[-30^o, 30^o]$. Table III summarizes the key steps of the RRT planner we implement.

TABLE III
PRIMITIVES-BASED RRT PLANNER STEPS

1. Read *Workspace*, q_0, q_d;
2. Read the S, L, and R primitives;
3. **for** $i = 1$ **to** k **do**
4. Sample random point $\alpha(i)$ in free workspace;
5. Find vertex q_n closest to $\alpha(i)$;
6. Create S, L, and R edges from q_n;
7. Add collision-free edges;
8. Update vertex list;
9. Exit if a neighborhood of q_d is reached;

In Fig. 7, the actual obstacles are marked in black, and light gray is used to denote the areas where the boundary of the robot[3] intersects with the boundary of an obstacle for any robot orientation. Curves in blue correspond to the branches of the constructed tree, and the sequence of STAR paths that lead from the initial configuration to the neighborhood of the desired configuration are highlighted in red. The output of the planner in terms of sequencing of motion primitives for each case is presented in Table IV. The complexity introduced by additional obstacles results in longer plans involving more switching among primitives.

TABLE IV
OUTPUT OF THE RRT PLANNER

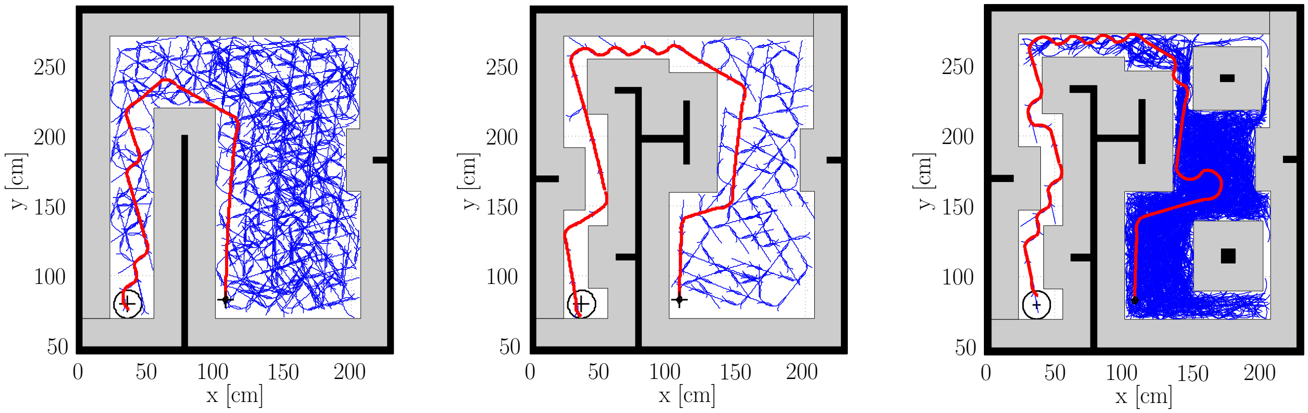
Case	Output
(a)	$\{S\ S\ S\ S\ S\ L\ S\ S\ L\ S\ L\ S\ R\ L\ S\ S\ R\ L\ R\ L\ L\}$
(b)	$\{S\ S\ R\ S\ L\ S\ S\ S\ L\ S\ S\ L\ R\ L\ R\ L\ R\ L\ L\ S\ S\ S\ S\ R\ S\ L\ S\ S\ L\}$
(c)	$\{S\ S\ R\ S\ S\ L\ L\ L\ R\ R\ S\ S\ L\ R\ L\ S\ L\ R\ L\ R\ L\ R\ L\ L\ S\ R\ L\ L\ R\ S\ R\ L\ R\ L\ R\ L\ S\ L\}$

IV. Experimental Results

In this section we report on experimental tests of the RRT planner implemented with the parameters of Table IV and applied on STAR. For each case of Fig. 7, we collect open-loop position and orientation measurements for a total of 30 trials. Just as in the calibration stage, all trials are

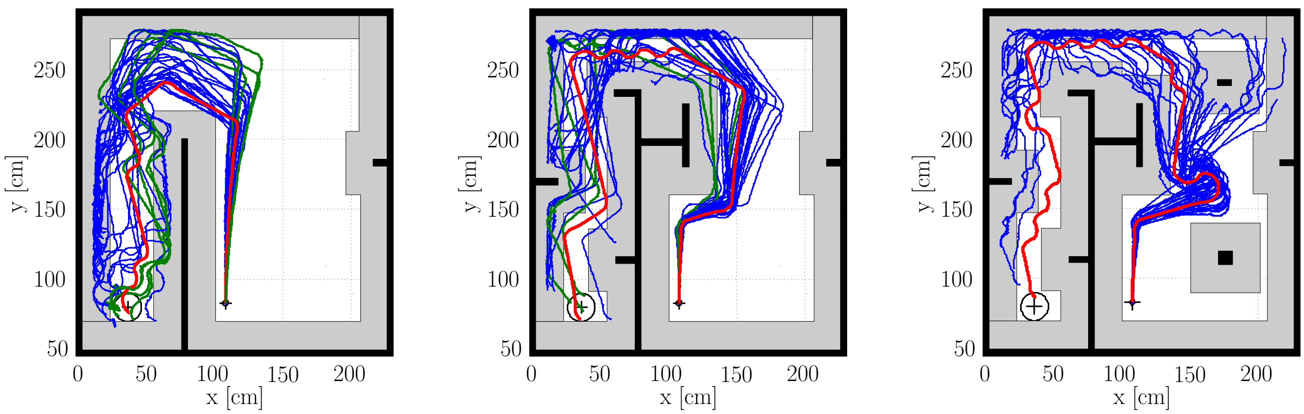
[2]The RRT is deemed sufficient as we consider a single initial-goal pair configuration. Cases with multiple initial-goal pairs are tackled by employing *probabilistic roadmaps* [30] (PRMs).

[3]At the particular sprawl angle considered, and including the length of the spokes, the robot has roughly the size of a square with side length equal to 18 cm.



(a) (b) (c)

Fig. 7. Implementation of an RRT planner in our particular problem. The generated tree begins from the initial configuration, and branches out incrementally until the desired configuration (marked inside the black circle to the bottom left corner) is reached. The map increases in complexity from left to right by adding more obstacles. (a) The basic map: Many solutions exist, and the resulted shortest path involves minimal switching among robot actions. (b) A set of obstacles has been added to block the initial path. The planner has to respect the motion constraints of the problem; this leads to the "wavy" motion pattern close to the top left corner. (c) The most complicated environment considered: Two areas to the right are now inaccessible; as a result, the plan requires a large amount of switching.



(a) (b) (c)

Fig. 8. Experimental implementation of the plans shown in Table IV. (a) Least complex workspace: 4 paths (highlighted in green) reach the desired configuration. (b) Medium-complexity workspace: 3 trials reached the goal. (c) No successful trials were recorder for the hardest workspace.

conducted on the same rubber floor mat surface, and the robot is manually set into the designated initial configuration; the initial pose errors are included in Table V.

TABLE V
ERROR IN INITIAL POSE (EXPERIMENTAL TESTING)

Case	Mean [cm, cm, deg]	Standard Deviation [cm, cm, deg]
(a)	(0.07, 0.11, 1.14)	(0.07, 0.09, 0.72)
(b)	(0.12, 0.11, −0.38)	(0.50, 0.61, 2.14)
(c)	(0.04, 0.10, 0.52)	(0.04, 0.08, 0.38)

Figure 8 presents the experimental results from the open-loop RRT plan implementation. Curves in red indicate the planned path. Curves in blue correspond to the actual experimental outcome. Uncompensated accumulated errors make actual paths deviate substantially from the planned. Curves in green mark the successful trials.

A large number of paths bring the robot in contact with the workspace boundary. In most of these cases, the robot was able to keep making progress toward its goal. Moreover, there exist cases where the robot-wall interaction was beneficial. For instance, Fig. 8(a) shows that after the impact, the wall compensated for the accumulated error, and aided STAR in moving closer to the goal. On the contrary, if no walls were present, the robot would have deviated significantly from its predetermined path.

With respect to Fig. 8, 95%, 83%, and 59% of paths for cases (a), (b), and (c), respectively, were implemented in full before a terminal contact with an obstacle or the boundary occurred. Similarly, 94.3%, 93.8%, and 75.9% of the planned path length was covered for each case. Finally, 13.3% of paths succeeded in reaching the goal for case (a), 10% for case (b), and there were no successful paths for the hardest workspace case of Fig. 8(c).

We expect that state feedback either from motion capture (ground truth) or on-board sensors (IMU) will increase the number of paths that succeed in reaching the target configuration; this is part of ongoing work.

V. Conclusions

The novel STAR robot offers promising mobility capabilities for navigating in a range of challenging environments. Due to its reconfigurable sprawl angle, STAR combines the benefits of wheeled and legged locomotion, and can adjust its performance depending on its environment. This feature can be exploited to tackle non-trivial motion planning problems.

This work presents the first implementation of motion planning techniques to STAR, and offers a first glimpse into the open-loop performance of the robot when executing autonomously a precomputed motion plan, in workspaces of increasing complexity. With the sprawl angle kept constant, and at the particular configuration considered, the robot's locomotion can be captured adequately by a (unicycle) Dubins vehicle. We construct three Dubins curves (straight line, left turn, and right turn) based on collected experimental data. These curves are then combined with a generic single-tree RRT planner to create motion plans that steer the robot from an initial to a desired configuration inside its workspace.

Our experiments show that the generated motion plan is initially followed by the robot, but as the accumulated error grows, the robot drifts away from the planned path, and comes into contact with the obstacles and workspace boundaries. Closing a low-level feedback loop will definitely be beneficial to planning accuracy. Yet, non-catastrophic collisions may actually turn out to be beneficial in terms of making progress toward the goal, as they may compensate for the accumulated error.

Ongoing work involves integrating state feedback through on-board sensing to follow predetermined motion plans, investigating the suitability of bio-inspired models to capture the robot's behavior in the horizontal plane, and experimenting on unstructured terrains by actively controlling the sprawl angle to overcome obstacles.

References

[1] D. Zarrouk, A. Pullin, N. Kohut, and R. Fearing, "STAR - a sprawl tuned autonomous robot," in *IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 20 – 25.

[2] A. Yumaryanto, J. An, and S. Lee, "A cockroach-inspired hexapod robot actuated by LIPCA," in *IEEE Conf. on Robotics, Automation and Mechatronics*, 2006, pp. 1 – 6.

[3] J. M. Morrey, B. Lambrecht, A. D. Horchler, R. E. Ritzmann, and R. D. Quinn, "Highly Mobile and Robust Small Quadruped Robots," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 1, 2003, pp. 82–87.

[4] B. Lambrecht, A. D. Horchler, and R. Quinn, "A small, insect-inspired robot that runs and jumps," in *IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 1240 – 1245.

[5] S. Kim, J. E. Clark, and M. R. Cutkosky, "iSprawl: Design and Tuning for High-speed Autonomous Open-loop Running," *The International Journal of Robotics Research*, vol. 25, no. 9, pp. 903–912, Sep. 2006.

[6] R. Merz, F. Prinz, K. Ramaswami, M. Terk, and L. Weiss, "Shape Deposition Manufacturing," in *Proceedings of the Solid Freeform Fabrication Symposium*, 1994.

[7] R. Wood, S. Avadhanula, R. Sahai, E. Steltz, and R. Fearing, "Microrobot Design Using Fiber Reinforced Composites," *Journal of Mechanical Design*, vol. 130, no. 5, pp. 1 – 11, May 2008.

[8] P. Birkmeyer, K. Peterson, and R. S. Fearing, "DASH: A dynamic 16g hexapedal robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 2683 – 2689.

[9] A. M. Hoover, S. Burden, X.-Y. Fu, S. Sastry, and R. S. Fearing, "Bio-inspired design and dynamic maneuverability of a minimally actuated six-legged robot," in *IEEE Int. Conf. on Biomedical Robotics and Biomechatronics*, 2010, pp. 869–876.

[10] D. W. Haldane, K. C. Peterson, F. L. G. Bermudez, and R. S. Fearing, "Animal-inspired design and aerodynamic stabilization of a hexapedal millirobot," in *IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 3279 – 3286.

[11] A. Pullin, N. Kohut, D. Zarrouk, and R. Fearing, "Dynamic turning of 13 cm robot comparing tail and differential drive," in *IEEE Int. Conf. on Robotics and Automation*, 2012, pp. 5086–5093.

[12] A. Mathis, J. Russell, T. Moore, J. Cohen, B. Satterfield, N. Kohut, X.-Y. Fu, and R. Fearing, "Autonomous navigation of a 5 gram crawling millirobot in a complex environment," in *Adaptive Mobile Robotics: 15th Int. Conf. on Climbing and Walking Robots and the Support Technologies for Mobile Machines*, 2012, pp. 121 – 128.

[13] D. Zarrouk, A. Pullin, N. Kohut, and R. Fearing. (2013) STAR - a sprawl tuned autonomous robot. Video submission for IEEE Int. Conf. on Robotics and Automation. [Online]. Available: http://www.youtube.com/watch?v=vXVRCpDLSHI

[14] D. Zarrouk and R. Fearing, "Cost of transport of a dynamic hexapedal robot," in *IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 2548 – 2553.

[15] W. J. Schwind, "Spring Loaded Inverted Pendulum Running: A Plant Model," Ph.D. dissertation, University of Michigan, 1998.

[16] I. Poulakakis and J. Grizzle, "Modeling and control of the monopedal robot Thumper," in *IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 3327–3334.

[17] J. Schmitt and P. Holmes, "Mechanical models for insect locomotion: dynamics and stability in the horizontal plane I. Theory," *Biological Cybernetics*, vol. 83, no. 6, pp. 501–515, Nov. 2000.

[18] ——, "Mechanical models for insect locomotion: dynamics and stability in the horizontal plane - II. Application," *Biological Cybernetics*, vol. 83, no. 6, pp. 517–527, Nov. 2000.

[19] P. Holmes, R. J. Full, D. E. Koditschek, and J. Guckenheimer, "The Dynamics of Legged Locomotion: Models, Analyses, and Challenges," *SIAM Review*, vol. 48, no. 2, pp. 207–304, 2006.

[20] J. E. Seipel, P. J. Holmes, and R. J. Full, "Dynamics and Stability of Insect Locomotion: A Hexapedal Model for Horizontal Plane Motions," *Biological Cybernetics*, vol. 91, no. 2, pp. 76–90, 2004.

[21] D. Jindrich and R. Full, "Many-Legged Maneuverability: Dynamics of Turning in Hexapods," *The Journal of Experimental Biology*, vol. 202, pp. 1603 – 1623, Jun. 1999.

[22] J. Proctor and P. Holmes, "Steering by Transient Destabilization in Piecewise-Holonomic Models of Legged Locomotion," *Regular and Chaotic Dynamics*, vol. 13, no. 4, pp. 267–282, 2008.

[23] D. Zarrouk and R. Fearing, "Compliance-based dynamic steering for hexapods," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 3093 – 3098.

[24] K. Karydis, I. Poulakakis, and H. G. Tanner, "A switching kinematic model for an octapedal robot," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 507–512.

[25] L. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, no. 3, pp. 497 – 516, Jul. 1957.

[26] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006.

[27] H. Choset, W. Burgard, S. Hutchinson, G. Kantor, L. E. Kavraki, K. Lynch, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementation*. MIT Press, 2005.

[28] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," in *IEEE Int. Conf. on Robotics and Automation*, 1999, pp. 473 – 479.

[29] K. Karydis, I. Poulakakis, and H. G. Tanner, "Probabilistic validation of a stochastic kinematic model for an eight-legged robot," in *IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 2562 – 2567.

[30] L. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars, "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566 – 580, 1996.

[31] S. Karaman and E. Frazzoli, "Sampling-based optimal motion planning for non-holonomic dynamical systems," in *IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 5041 – 5047.