Shady3D: A Robot that Climbs 3D Trusses

Yeoreum Yoon and Daniela Rus

Abstract— This paper describes a truss climbing robot we designed and prototyped. The robot has a minimalist design with three motive degrees of freedom that enable movement along three-dimensional truss structures. This robot can form a six-degree-of-freedom structure by connecting to another identical module using a passive bar as a medium. We present the design and implementation of this robot, control algorithms for moving the robot in a 3-D truss structure, and hardware experiments.

I. Introduction

We wish to create simple and robust autonomous robots capable of climbing. In this paper we focus on robots capable of climbing 3D trusses, as a step toward tree-climbing robots. A truss is a structure composed of many straight, rigid bars connected together at structural nodes. This type of object is common in many large civil and industrial structures (e.g. bridges, towers, communication antennas, and construction scaffolds) and space structures (e.g. space station components and solar panel supports.) Tasks related to trusses are often dangerous or difficult for human workers. Because bars in truss structures are narrow, workers on high towers or construction scaffolds have more risk of falling than those on wide and flat floors. Space construction and maintenance outside a spacecraft require dangerous extravehicular activity (EVA) missions by astronauts. Truss-climbing robots have significantly more robustness challenges than wall climbing robots. They will move like acrobats across linkage structures, replacing or assisting humans in dangerous activities.

In this paper we describe the robot Shady3D (see Figure 1), a 3-dimensional truss-climbing robot. The robot consists of one structural bar with two rotating grippers at each end. The bar has an additional rotational degree of in the middle. These three rotational degrees of freedom of the robot are sufficient for locomotion on 3D trusses. This robot extends the robot system Shady2D described in [1], [2] by adding one degree of freedom, reducing significantly the size and redesigning the grippers to suit the 3D locomotion task.

Shady3D is a small robot of length 250mm and width 80mm. It moves within a plane by using its ability to grasp and pivot. It makes transitions between planes by using the rotational degree of freedom on the bar. Proprioception and environmental sensors are used to cope with misalignment errors and guarantee each attempted grasp on the truss.

We present the hardware design for this robot, the suite of control algorithms used for moving on 3D trusses, and results from extensive hardware experiments with 2 robots.

A. Related Work

Mobile robots that climb and traverse trusses have been investigated previously [2]–[6]. Amano et al developed a handrail-gripping robot for fire fighting [4]. Balaguer et al present a climbing autonomous robot that can move in complex 3-D metallic-based structures such as bridges [7]. Nechyba et al's SM2 robot was designed to walk along the I-beam truss structure of a space station [8], [9]. Although these truss climbing robots vary in locomotion and gripping mechanisms according to their specific applications, they have a common characteristic in that all degrees of freedom to perform a full range of required locomotion are implemented in a single robot. For example, Balaguer et al's robot has six motive degrees of freedom to reach an arbitrary pose in 3-D space [7].

Truss climbing is a special type of structure climbing. Other types of climbing includes wall climbing. Pack et al [10] present an inspection robot that can climb a 2-D planar surface. Two vacuum fixtures were employed as an attaching mechanism. Bretl et al [11] developed a free-climbing four-limbed robot, named Legged Excursion Mechanical Utility Rover (LEMUR). This robot was designed to climb a vertical wall that has pegs.

II. Shady3D Hardware

Shady3D is capable of autonomously grasping and moving on trusses. It moves by grasping a truss element and using its rotating degrees of freedom to pivot about the grasping point. The robot is a 3D extension of the two-dimensional Shady window shading robot [2].

The overall shape of the Shady3D robot resembles a stick (See Figure 1). It is composed of two rotating *grippers* linked by a two-part *arm*. Each gripper can grasp and release a truss bar by closing and opening its paddles. Each gripper is connected to the arm by a rotating joint (the gripper joint), which enables the gripper to align with a truss in various orientations. The two sides of the arm are connected by another rotating joint (the middle joint), which creates a relative angle between the directions of the two grippers. Shady3D has five degrees of freedom: three rotational degrees of freedom for locomotion (actuated by MicroMo 1724TR00SR DC micromotor combined with a 66:1 gearhead) and two degrees of freedom for gripper opening and closing (actuated by Sanyo NA4S mini gearmotors with 298:1 gearhead.) The robot includes proprioception sensors and environmental sensors, on-board computation (provided by the Gumstix Linux miniature computer, Robotstix board, and custom-built motor control boards), and on-board power (provided by four 3.7V, 750mAh Polymer Li-ion batteries). The robot's
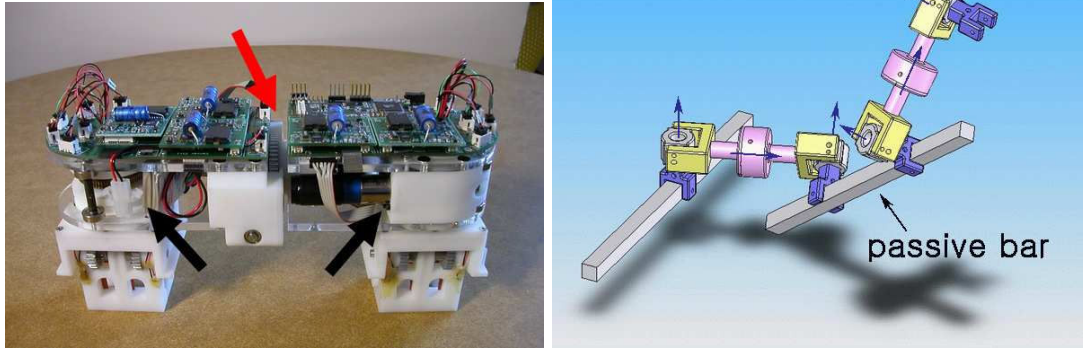
Fig. 1. (Left) Shady3D robot. The degrees of freedom of the grippers are indicated by the black arrows. The degree of freedom of the middle joint is indicated by the red arrow. (Right) A 6-DOF manipulator composed of two 3-DOF Shady3D modules and a passive truss element. Note that the axes of the two grippers holding the passive truss are not on the same line.

dimensions are $250mm \times 80mm \times 133mm$ and its weight is 1.34kg. All the structural parts for Shady3D are rapid prototyped in plastic or consist of PCB boards.

A. Grippers

The grippers are the main enablers of the robot's locomotion. In order to achieve reliable movement, the gripper must achieve a firm hold on the grasping surface to avoid slippage and falling off. Thus, reliability is the main design consideration for the gripper. Other important design goals include compliance and error tolerance.

We designed a gripper that meets these design requirements. The gripper was designed to envelope firmly a truss element with a $\frac{3}{4}in \times \frac{3}{4}in$ square cross section. When it is closed onto a truss, the four faces of the truss are fully encompassed by the gripper paddles and the contact surface of the housing. This design prevents the robot from wobbling on the truss. In addition, rubber pads were attached on the contact surface of the gripper paddles to increase friction and prevent slippage.

Each gripper structure is composed of four main components–the housing, the housing cover, the housing top cover, and two gripper paddles. The gripper housing forms the cubical gripper shape and supports shafts on which gears for the gripper mechanism are assembled. It also provides a contact surface for the truss. The housing cover and the housing top cover have mounting places for the gripper motor. The motor is mounted between these two covers. A worm gear used for the gripper joint mechanism is also placed between the two covers. Each cover has a groove for gripper joint bearings around its circumference.

Several features are implemented to accomplish compliance for misalignment and tilt. To correct misalignment four detector switches were installed on the gripper paddles. The states of these switches are checked during the grasping procedure. If the gripper is not aligned with the truss, one switch or two switches in diagonal position are pressed before the others. Based on this information, the corresponding gripper joint is rotated to resolve the misalignment. Compliance for tilt caused by gravity is accomplished mechanically with
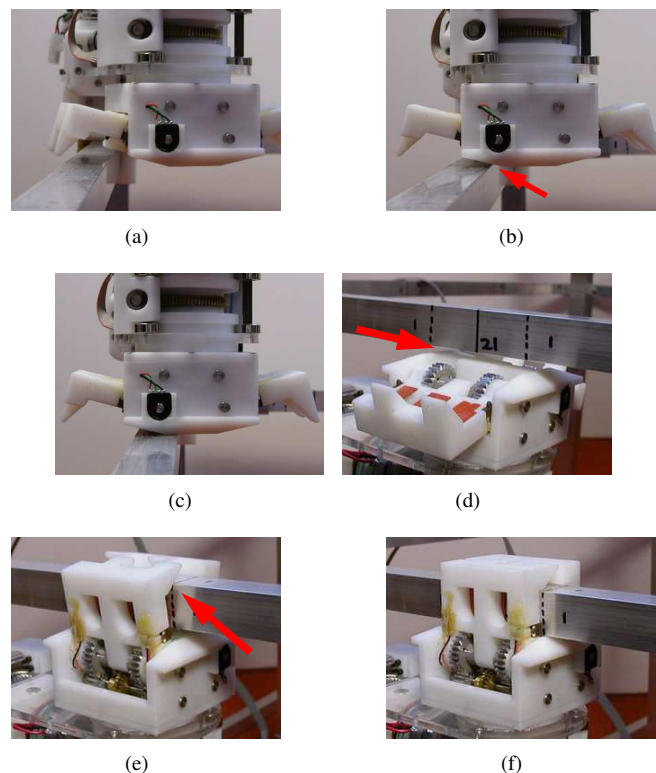

Fig. 2. The compliance of the gripper for tilt. (a)-(c) The slope in the housing, marked with the red arrow in (b), guides the gripper along the edge of the truss. (d)-(f) The slopes of the gripper paddles (highlighted in (e)) help the gripper pull the truss and compensate the gap (marked in (d)) between the contact surface and the truss.

a slope along the edges of the housing and slopes in the gripper paddles. When the gripper paddles are closed, the truss is guided and pulled to the contact surface by the slopes (see Figure 2). are rotated around the shafts mounted on the housing, and the range of rotation is about 100 degrees. Because of this wide range of rotation, the gripper paddles can be retracted behind the contact surface of the housing. This allows an open gripper to move over a truss without collision.

The actuation of the gripper was implemented with gear power transmission. Opening and closing the gripper is achieved as follows. The motor is installed vertically between two housing covers. A worm connected to the motor shaft drives two worm gears for both grippers. Each worm gear transmits power to small spur gears assembled on the same shaft. The small gears, in turn, drive engaged large spur gears, which are fixed to the gripper paddles. Since the gripper paddles are fixed to the large spur gears, they rotate at the same rate as the large spur gears. Because power transmission routes for the two paddles are symmetric both grippers move symmetrically like a mirror image.

B. Sensing

Shady3D uses sensors to estimate its own configuration, sense the environment (e.g. the truss bars), and to cope with misalignment error. The three joint motors have 512 counts/rev encoders which are used for controlling the joint to reach a desired angle. To sense the state of the gripper, a potentiometer connected to one of the gripper paddle shafts is used. Four detector switches are incorporated in each gripper for misalignment detection. One switch is mounted on the side of each gripper paddle. Information on misalignment can be obtained by checking the state of these four switches during the closing procedure of the gripper. If the gripper is not properly aligned along the truss, one switch or two switches in diagonal position are triggered before the others. We can adjust the corresponding gripper joint on the basis of this information. These switches can be used to detect misalignment in both the horizontal and the vertical directions.

C. Electronics and Control

Shady3D's electronics system consists of three layers: the motor control boards, the Robostix control board, and the Gumstix miniature linux computer. The motor control boards and Robostix are used for low-level motor control. The Gumstix computer is used for high-level control and planning. These components are part of the robot structure as shown in Figure 3.

Sensor information is transferred from the low-level layer (the motor controllers) to the high-level layer (the Gumstix or the workstation). Five motor controllers collect information from the sensors and send this information to the Robostix microcontroller. The Robostix controller organizes these sensor values and sends them to the Gumstix computer or to the workstation via serial communication. The control program running on the highest layer evaluates the state of the robot according to these sensor readings.

The control commands flow from the high level to the low level. To perform basic motion primitives, the control program sends necessary commands to Robostix. According to the commands, the Robostix controller selects an appropriate motor controller and transmits commands to it. Then, the motor controller generates a PWM signal to drive the motor. This basic strategy is used to implement the motion primitives used by this robot.

The robot's basic motion primitives are as follows:

- **Open Gripper.** This motion primitive opens the paddles of the designated gripper so that it can release a truss. After it is performed, the gripper paddles are fully retracted behind the contact surface of the gripper housing to prevent collision while the gripper moves over a truss. The average time for this operation is 20.8 sec.
- **Close Gripper.** This motion primitive ensures a reliable grasp on a truss. A major challenge is the correction of gripper misalignment. Failure detection and recovery is done using four detector switches mounted on gripper paddles and the corresponding joint. The average time for completing this operation is 24.0 sec.
- **Joint Rotation.** The rotating motion is implemented in two ways: (1) rotating a joint *by* a given angle, and (2) rotating a joint *to* a desired joint angle (goal position). Rotations are parameterized by the joint performing the rotation. The gripper joints provides locomotion capabilities within a plane. The middle joint rotation enables the robot to swap motion planes. The average time for completing a 90 degree rotation is 14.4 sec.
- **Joint Absolute Position Referencing.** The position referencing switch for the gripper is triggered when the gripper is in the zero-joint-angle position. The zero-joint-angle position is defined as the state where the gripper is aligned with the body line of the robot. For the middle joint, the switch is triggered when the axes of the gripper joints points toward the same direction. The implementation is as follows. The joint is rotated in one direction until the referencing switch is turned on, and then the join motor is stopped and the counter of the motor is set to zero.

III. PLANNING ALGORITHMS

Shady3D computes its best path to a desired next location on the truss using a planning algorithm. The main challenge for planning is to incorporate path length as well as the cost of difficult maneuvers (e.g. switching planes) in the optimization. Because the truss elements have square cross sections, the robot may be located in several configuration along each truss element. Thus, we need to incorporate the relative orientation of the robot on a truss element in the state representation.

We assume that the model of the truss is given. We also assume the robot knows its current position on the truss.

An abstract graph of the environment is used for path planning. The graph is composed of *nodes*, which represent discrete gripping points on trusses, and *edges*, which represent connection between two nodes. The robot is located at one of the nodes. Given a goal location, the robot computes the path in the graph from its current node to the goal node. The path becomes a sequence of nodes for the robot to follow.

More precisely, a node represents a gripping point. It is specified by three elements: the position of the node, the direction of the node on the truss, and the face of the node. Node direction and face are needed in order to determine the orientation of the robot on a truss edge. Since each truss
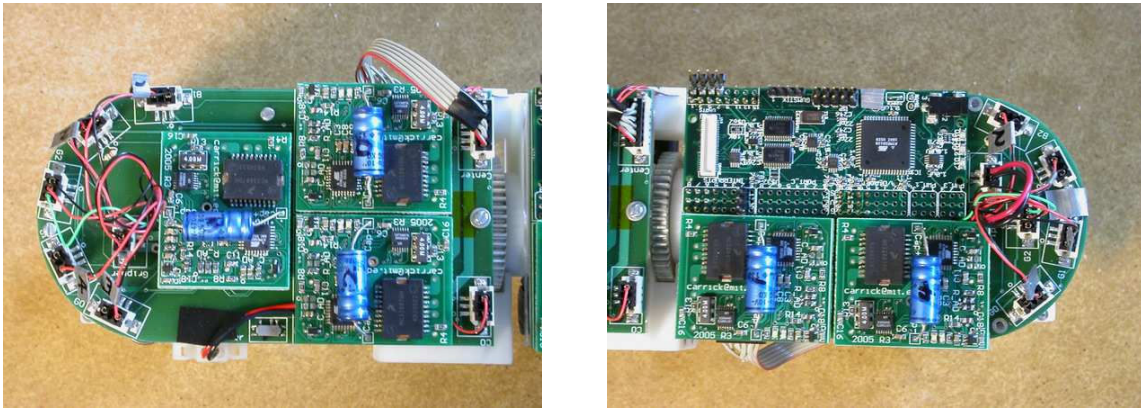
Fig. 3. This figure shows the circuit boards of the Shady3D robot. The left image shows the circuit board on the left side, containing three motor control boards. The right image shows the circuit board on the right side, containing two motor control boards and the Robostix board. The Gumstix computer is removed to show the Robostix board clearly.

edge is a parallelepiped, the robot can be positioned on an edge in 8 different ways.

The *path* of the Shady3D robot is a sequence of nodes to be followed successively. The path planning algorithm determines a sequence with minimum cost among them.

The *cost* optimization of the path goes beyond the physical length of the path. The most difficult robot maneuver is the transition to a different plane of motion. Therefore the number of plane transitions is also included in the cost calculation. Note that even though the physical lengths of two paths is the same, their cost can be different. Figure 4 shows an example. In this figure, two different paths from the start node to the goal node are shown. The physical lengths of Path A and Path B are the same. However, while Path A includes one transition between different planes (indicated by the red-shaded circle in Figure 4(a)), Path B requires two transitions between different planes (indicated by the red-shaded circles in Figure 4(b)). Transition between different planes requires the cooperation of two Shady3D robots. Such cooperation takes more time and computation and consumes more energy than the step movement of an individual robot. Therefore, Path B can be considered more difficult, or more costly, though its physical length is the same as that of Path A.

The path planning algorithm evaluates cost of paths in terms of both the length and difficulty of movement, and finds the most efficient path with the minimum cost. It uses Dijkstra's shortest path algorithm on the environmental graph. Because edges connecting two nodes that can be traversed only by the cooperation of two robots have larger cost (multi-cost) than those connecting two nodes reachable by an individual robot have (single-cost), Dijkstra's algorithm can compute the most efficient path that includes a minimum number of inter-plane transitions.

IV. SHADY3D EXPERIMENTS

Experiments with Shady3D hardware were performed in a custom-designed truss environment built with 3/4-inch-wide square-cross-section aluminum tubes (see Figure 5).
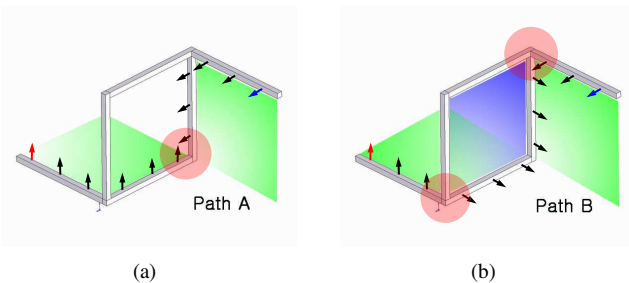

(a)　　　　　　　　(b)

Fig. 4. This figure shows two paths from the start node (the red arrow) to the goal node (the blue arrow). The nodes are represented by their normal vectors. The physical lengths of the two paths are the same, but the cost is different. The shaded circles in red represent inter-plane transition that requires two-Shady3D cooperation. Planes covered by each path are highlighted in green and blue.

The truss is a rectilinear 3-dimensional structure and includes a main square horizontal frame with a vertical frame built inside. The vertical frame is located so that an individual Shady3D robot can move from the horizontal frame to the vertical frame with a single step. The horizontal frame is supported by four legs.

The nodes, which are spaced by the center-to-center distance (180mm) of the robot, are marked on the positive faces of the trusses. Each truss in the horizontal frame has eight nodes—four for each positive face—so the number of nodes in the horizontal frame is 32. The vertical frame contains 14 nodes, and four leg trusses have eight nodes in total. Thus, the total number of nodes in the environment is 54.

We have tested hundreds of single step and multiple step navigation for Shady3D within this environment.

The single step move of the Shady3D robot was tested for various situations as follows:

- Straight movement in a horizontal plane.
- Transition between trusses in a horizontal plane.
- Straight movement in a vertical plane, in the horizontal direction.
- Straight movement in a vertical plane, in the vertical

Fig. 5. This figure shows the Shady3D testing environment. The origin is the left-rear corner of the horizontal frame in the image (marked by the XYZ frame). Indices and gripper boundaries are marked for the nodes. The vertical frame in the middle is located so that an individual Shady3D robot can move from the surrounding horizontal frame to the vertical frame by itself.

direction.
- Transition between trusses in a vertical plane.
- Transition between a horizontal plane and vertical plane.

The multi-step navigation was tested with different locations for the starting and goal configuration. The planner was used to compute the best path and Shady3D executed the path. Figure 6 shows some snapshots from such an execution. 15 different test configurations have been tried multiple times each. The shortest path was 4 steps long, the longest path was 10 steps long, and the average path consisted of 8 steps. Many paths included transitions from the horizontal structure to the vertical structure. The overall observed success rate was 95%, with all the errors due to gripper malfunction.

Table I summarizes the results for the trials we recorded for evaluation purposes (many more successful trials have been done as part of demos). Most failures occurred during transitions between trusses in a vertical plane (between Node 35 and Node 36). Except for the case where the mechanical malfunction of one gripper switch caused a failure, the original cause for these failures was the error in the anchor joint and consequent downward tilt of the robot body. Particularly, the downward tilt occurred during transitions from the horizontal frame to vertical frame, which was not compensated, was the main cause of the problems. Though such errors were overcome in most cases, this problem needs to be handled for more robust locomotion.

V. DISCUSSION

In this paper we describe the design of a 3-D truss climbing robot. The design follows a philosophy of minimalism and achieves both simplicity and locomotion capability. The key challenge in designing this robot has been to provide robustness for grasping the truss. Grasping a narrow truss element repeatedly is challenging as error tends to accumulate. Compliance and sensing were used for error detection and recovery.

The hardware experiments in our test environment showed that the design and algorithms of the Shady3D robot work successfully and reliably for navigation in the 3-D truss structures. However, some shortcomings that require further improvement have been found, particularly in the hardware design. The main hardware failure we observed during the experiments was in the gripper joint. This error resulted from the bearing joint mechanism used for the gripper joint. The gripper joint was particularly weak for the axial load, which had not been considered seriously in the design stage. In the future we will use a better mechanism. We have also learned that the Gumstix software is not sufficiently reliable. We had to debug and code several extensions to the Gumstix Java and ultimately found that Gumstix is not reliable when used in the context of high-speed motors.

Our future goals for this system are in cooperative construction between multiple robot units and passive truss elements. We wish to accomplish robot systems in which the robot elements assemble the truss, with the robots comprising the joints of the truss.

VI. ACKNOWLEDGMENTS

REFERENCES

[1] C. Detweiler, M. Vona, K. Kotay, and D. Rus, "Hierarchical control for self-assembling mobile trusses with passive and active links," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, 2006.

[2] M. Vona, C. Detweiler, and D. Rus, "Shady: Towards robust truss climbing with mechanical compliances," in *Proceedings of the 2006 International Symposium of Experimental Robotics*, 2006.

[3] K. D. Kotay and D. L. Rus, "Navigating 3D steel web structures with an inchworm robot," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 1996.

[4] H. Amano, K. Osuka, and T.-J. Tarn, "Development of vertically moving robot with gripping handrails for fire fighting," in *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, Hawaii, USA, 2001, pp. 661–667.

[5] R. Aracil, R. J. Saltarén, and O. Reinoso, "A climbing parallel robot," *IEEE Robotics and Automation Magazine*, pp. 16–22, Mar. 2006.

[6] A. Greenfield, A. A. Rizzi, and H. Choset, "Dynamic ambiguities in frictional rigid-body systems with application to climbing via bracing," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, Apr. 2005, pp. 1959–1964.

[7] C. Balaguer, A. Giménez, J. Pastor, V. Padrón, and M. Abderrahim, "A climbing autonomous robot for inspection applications in 3d complex environments," *Robotica*, vol. 18, pp. 287–297, 2000.

[8] M. C. Nechyba and Y. Xu, "SM2 for new space station structure: Autonomous locomotion and teleoperation control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, May 1994, pp. 1765–1770.

[9] ——, "Human-robot cooperation in space: SM2 for new space station structure," *IEEE Robotics and Automation Magazine*, vol. 2, no. 4, pp. 4–11, Dec. 1995.

[10] R. T. Pack, J. L. C. Jr., and K. Kawamura, "A rubbertuator-based structure-climbing inspection robot," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Albuqurque, New Mexico, Apr. 1997.
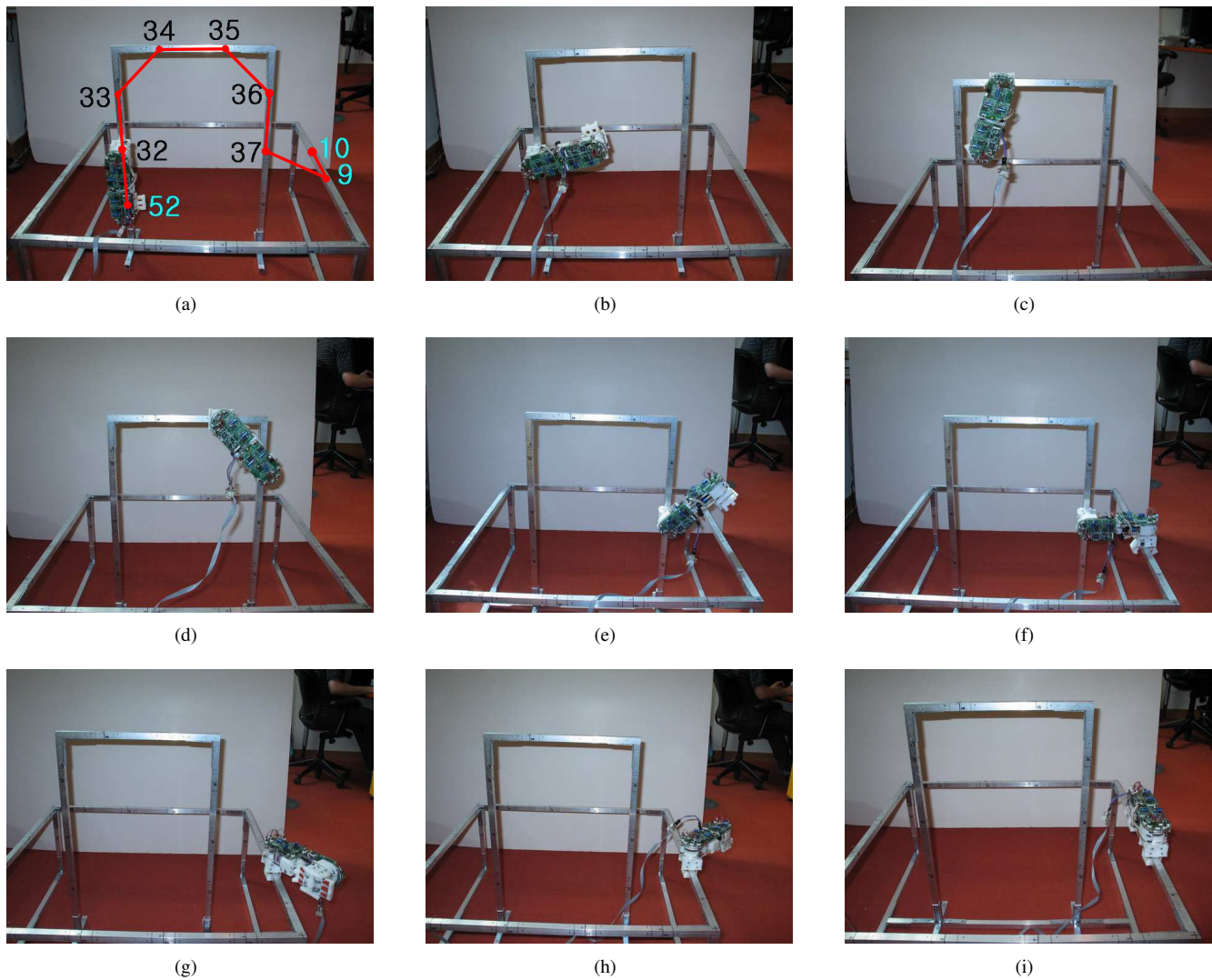
(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

(i)

Fig. 6.   Nine snapshots of navigation. The robot computed and followed the most efficient path (highlighted in (a)).

Type	Attempted	Successful	Success Rate	Failure Modality
straight, horizontal plane	60	60	100.0%	
transition, horizontal plane	17	15	88.2%	joint motor failure; imcomplete grip
straight, vertical plane, horizontal direction	30	28	93.3%	excessive current in gripper motor; larger gripper misalignment
straight, vertical plane, vertical direction	32	32	100.0%	
transition, vertical plane	21	17	81.0%	gripper switch malfunction; excess current in gripper motor; incomplete grip
transition between horizontal and vertical plane	16	16	100.0%	
Total	176	168	95.5%	

TABLE I

THE SUMMARY OF THE RESULTS OF HARDWARE EXPERIMENTS. ALL STEPS MADE IN EXPERIMENTS ARE CLASSIFIED ACCORDING TO THEIR TYPE.

[11] T. Bretl, S. Rock, J.-C. Latombe, B. Kennedy, and H. Aghazarian, "Free-climbing with a multi-use robot," in *Proceedings of the International Symposium of Experimental Robotics*, 2004.