A Two-DOF Laminate Walking Robot for Use in Research and the Classroom

Benjamin D. Shuch
Graduate Student (Graduated)
Ira A. Fulton Schools of Engineering
Arizona State University
Tempe, Arizona

Taha Shafa
Graduate Student
Ira A. Fulton Schools of Engineering
Arizona State University
Tempe, Arizona
Email: tshafa@asu.edu

Eric Rogers
Student (Graduated)
Ira A. Fulton Schools of Engineering
Arizona State University
Tempe, Arizona

Daniel M. Aukes
Assistant Professor
Ira A. Fulton Schools of Engineering
Arizona State University
Tempe, Arizona
Email: danaukes@asu.edu

In this article we present a low-cost, two degree-of-freedom laminate robot transmission for legged locomotion applications. This transmission is specifically applied in the design of a quadrupedal robot, and has the potential to be used in other multi-legged systems. It offers a complex control space with a variety of different programmable gait trajectories, while leveraging low-cost linkages made using laminate approaches. The two-degree-of-freedom kinematics of the leg are subsequently modeled in Python, and the workspace of the robot is then experimentally verified on an initial quadrupedal design. Critical design considerations include the laminate design, the rigidity of the materials that make up the laminate, and the range of motion the device can undergo.

1 Introduction

Laminate manufacturing techniques, inspired by origami and kirigami, have enabled the rapid creation of inexpensive, yet kinematically-complex robots. This differs from traditional rigid mechanical designs in which rigid and precise metal links are assembled into jointed systems. These robots are made using a rapid prototyping strategy which involves the iterative cutting and lamination of thin materials such as cardboard, fiberglass and plastic to create rigid links attached to each other via soft, flexure-based hinges [1] [2]. Using an analytic framework which permits the computation of manufacturing geometry from layered designs [3] [4], design concepts can be rapidly converted to viable designs. The high speed, low cost nature of this manufacturing method makes it feasible to fabricate many design iterations for experimentation and testing. This functionality has been made available in popupCAD [3, 4], a Python-based design tool that provides users with the capability to design laminates more efficiently. Overall, the robots produced with this method are low-cost and can be produced with a laser cutter and 3D-printer. The dynamics of laminate systems can also be simulated to a high degree of accuracy before manufacturing to ensure these robots will behave in a predictable manner [5, 6].

Some examples of the robots that have been made using laminate techniques are HAMR[7],[8], DASH [9], DynaRoACH [10], and the Flying Monkey[11]. HAMR, a 1.27 gram, 4 centimeters-long robot, uses two piezoelectric actuators at each of its legs to control its motion, and is capable of moving at a speed of up to 0.44m/s, or 44 body lengths per second along desired trajectories. It is expensive to make (due to the materials and tools required), and can carry a payload of 1.35 grams. DASH is an example of a hexapedal laminate robot that uses two motors to provide forward locomotion and turning. It is fast, small, lightweight, and uses complex and mechanically constrained mechanism that allows it to create bio-inspired motion that mimics a cockroach. DynaRoACH is a hexapod robot that relies on parallel kinematic mechanisms fabricated through laminate manufacturing processes. It uses SMA actuator wires and a motor to move; it weights 24g and is 10cm long. Its leg mechanism is mechanically constrained as well; the derived OpenRoACH design is also available as a low-cost, open-sourced design[12]. The Flying Monkey uses mechanically-integrated four-bar mechanisms and quad-rotor propellers in order to turn on the ground, as well as to fly. It weights about 30 grams and its size is about 6x6cm. Other than HAMR (with its limited payload), the drive mechanisms of these robots connect multiple feet together to provide a pre-programmed gait which is determined during the design. One exception is the C-Turtle [13, 14], a laminate turtle-inspired robot with two degrees of freedom in each flipper. Machine learning was used in conjunction with computer vision to identify an optimal gait for forward locomotion both in the lab and outdoor settings.

Outside of laminate robots, robots such as Big Dog [15], StarlETH [16], and Cheetah Cub [17] have complex mechanical work spaces. Big Dog has a variety of onboard systems that power, control and sense throughout the robot. It is able to move over many different types of terrain, and is an incredibly high functioning robot, but costs around 32 million dollars to develop [18]. StarlETH has controllable system torque which is used to complete highly dynamic maneuvers. These robots are effective for gait and motion analysis, but are expensive to manufacture and operate, and less accessible to research labs. Cheetah-cub is a lower-cost, animal-inspired robotic platform which has been utilized in the past for controls research because it offers an interesting quadrupedal platform at a more reasonable cost. Previous work included control methods using a central pattern generator open loop setup with no external sensing applied.

From a survey of these robots, we have identified an opportunity to provide a cm-scale, multi-legged, robotic platform for studying high degree-of-freedom locomotion and control at a target cost of $200. This price is more compatible with student research and swarm applications and is attainable by using laminate techniques to lower the manufacturing and material costs. Affordable manufacturing techniques allowed for multiple iterations of this robots design that can be seen in Figure 1. This robot's manufacturing and fabrication cost makes it ideal for all labs and classrooms regardless of funding, so to make this platform accessible to classrooms and research labs, the parts list, CAD files, and programming files are shared, along with instructions on how to assemble the quadruped [19].

Table 1. Lengths of linkages in Figure 2

Linkage	Length (cm)	Linkage	Length (cm)
l_1	2.5	l_7	6
l_2	3.5	l_8	4
l_3	4	l_9	3.5
l_4	6	l_{10}	2.5
l_5	6	l_{11}	7
l_6	6	l_{12}	1.5

The rest of the paper is organized as follows. Section 2 provides an overview of the device. Section 3 discusses the kinematics selected, while Section 4 discusses the mechanical design iterations from a physical perspective. Section 5 discusses the manufacturing planning and section 6 discusses mechatronics integration. Section 7 discusses a set of experiments performed on the quadruped and compares it to the modeled kinematics. Section 8 discusses current work being done by the robot and its applications. Section 9 discusses the future goals and applications of such a device.

2 Device Overview

A single leg of this robot consists of three four-bar linkages connected in a parallel-series configuration as seen in Figure 2. This system is actuated by two servo motors fixed to the body and connected to two links l_1 and l_{10}. The output four-bar linkages, consisting of links l_1, and l_2, connect to the body link l_4, a distance l_3 away on the left side. The other output four-bar linkage, consisting of l_9, and l_{10}, is connected to the body link l_7 a distance l_8 away on the right side. These proximally-located four-bar linkages serve to connect two input actuators to a foot connected via a third, distal four-bar linkage, consisting of links l_4, l_5, l_6, and l_7, for a total of two degrees of freedom per leg. Finally, l_{11} is a constrained value that provides the distance between the two servos in the X axis, and l_{12} is a constrained value that provides spacing for the servos to be mounted beside each other. The lengths of the links and position of the servos have been carefully selected using this kinematic model to maximize the end effector's (the foot's) range of motion while avoiding mechanism singularities and part interference. The values for those lengths can be found in Table 1.

Off-the-shelf components such as servos are attached with modular 3D printed parts, which are designed to interface with the laminate system, taking care to align servo motor axes with the moving joint axes to which they are attached. Focus has also been paid in the design to ensure that links are constructed or reinforced with stiff materials such


(a) (b) (c)

Fig. 1. Evolution of quadrupedal laminate robots made with discussed transmission

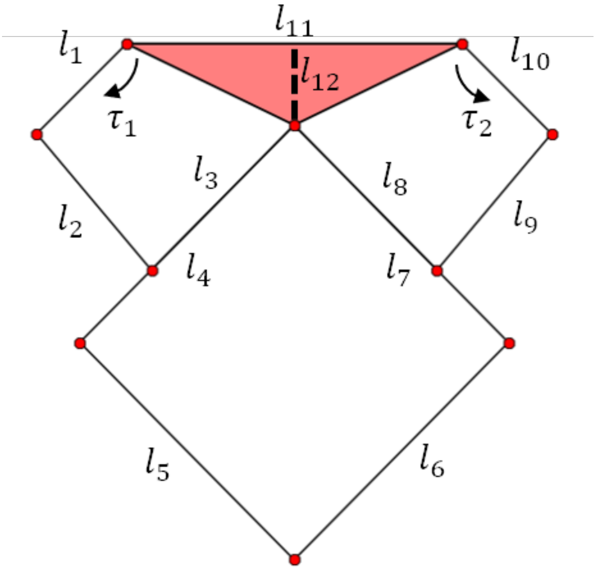

Fig. 2. Sketch of device with lengths labeled. The red area indicates points fixed to the body frame.

as fiberglass, in order to improve the lifespan of the transmission under payload and reduce unwanted flexibility.

3 Kinematics

The two-dimensional trajectory of the leg was modeled in Python using libraries such as *numpy* and *scipy*. This allows us to visualize how altering design parameters, such as link lengths and actuator locations, can result in useful changes in the foot's range of motion. Constraint equations defining the lengths of links and actuator angles were fed into a nonlinear minimization function provided by *scipy*, with an initial guess for the position of each vertex. Once a valid solution was obtained, the process was repeated throughout a range of servo positions in order to plot an entire cycle of the motion path.

Solving for the output trajectory requires that the positions of the actuators are specified together as a function of time. In order to minimize the number of control parameters required to describe this motion, we have used sine functions with a time offset to define the cyclical motion in the actuators according to

$$\theta_{i1} = A_{i1} * \sin(\omega_i t - \phi_{i1}) + o_{i1} \tag{1}$$

$$\theta_{i2} = A_{i2} * \sin(\omega_i t - \phi_{i2}) + o_{i2}, \tag{2}$$

where i indicates the leg number, A_{i1} and A_{i2} indicate the amplitude of each servo, ω_i is the (coupled) frequency of the two actuators, ϕ_{i1} and ϕ_{i2} are the time offsets in the sine function, and o_{i1} and o_{i2} are offsets that correctly align the output of the motor shafts with the linkages l_1 and l_{10}, as seen in Figure 2. This method of parameterizing our control signal gives us seven variables with which to create a cyclical trajectory instead of a infinite control space. Other general time-based signals could be applied to the model in the future.

A large variety of trajectories may be created with the same control law by modifying the lengths of the linkages, the range of angles they can cycle through, and their offset in the sine wave cycle. Figure 3 displays a variety of such patterns generated on the same leg design.

This model allows us to visually identify and debug undesirable leg trajectories, singularities, part interference, and other issues that can occur in a motion trajectory. One of the primary goals of the model was to increase the range of the servo actuators in each leg to maximize the range of motion of each servo. For example, in the early stages of the design, servo 1 could move 120°without interfering with other parts, but servo 2 could only move around 15°. By modifying the design using the simulation as a guide, the servos were subsequently able to create similar output paths while using a wider range of motion, effectively reducing the torques on each motor for the same end-effector loads.

While the solver is fairly effective at modeling the motion of the leg, it can fail when the device (or its description) moves through a singularity, if the step size between actuator positions grows too large, or if the initial input approximations are too far away from a valid solution. Attention must be paid by the designer during this process to validate that the solver is computing the device as intended.

4 CAD Design and Topology Exploration

Different layouts of the same mechanism have been investigated throughout the course of this project. The mechanism was originally designed using a 5-layer laminate with

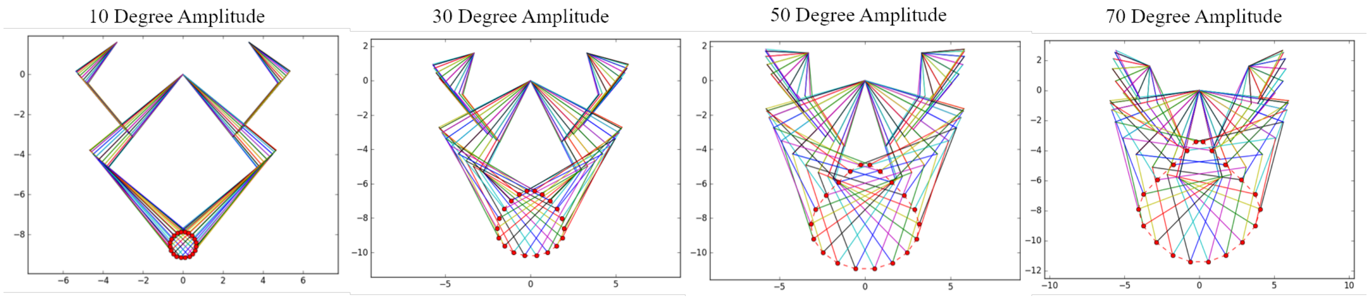

Fig. 3. Python kinematics simulation with angles that cycle between 10°-70° amplitudes and with a 90° phase offset

linkages arranged adjacent to each other in plane (Figure 1(a)). When compared to a modular mechanism made out of stacks of 5-layer laminates arranged with neighboring four-bar linkages stacked on top of each other (implemented in Figure 1(b,c)), we determined that using this alternate topology of the same kinematic device would provide better structural stability. This particular layout, as detailed in Figure 4, is made out of multiple separate 5-layer laminate parts that are connected using screws or plastic rivets. Each leg is symmetric along its own sagittal plane, with the motors mounted directly overhead. This allows the forces and torques transmitted through the linkage to be balanced evenly. This design was selected for its superior stiffness and because it helps to eliminate unwanted torsional effects at the joints.

The current, fully labeled design can be seen in Figure 4. The base is connected to the servo mounts and the custom leg connector. Servos are placed in the mounts and connected to servo horns. The laminate pieces are individually manufactured using laminate fabrication techniques (described in Section 5) and then assembled. There are five laminate parts in total, the main four-bar leg, two parts that connect the servos to the main four-bar leg, and the two connectors that hold the four-bar leg to the base. A single point of the four bar remains stationary and is mounted using the custom leg connector. By using modular laminate and parallel linkages the motors may be placed into a rigid frame or body, reducing leg mass and permitting faster accelerations of the foot.

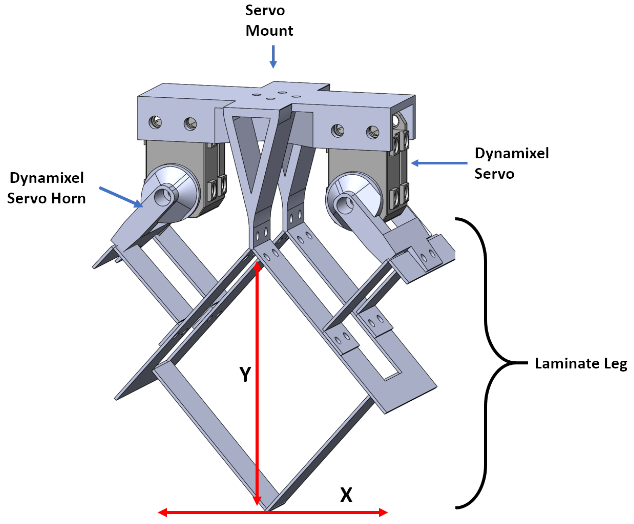

Fig. 4. Labeled Dynamixel servo leg transmission with 3D printed parts and laminates

5 Laminate Manufacturing

Laminates allow the use of non-traditional hinges to be quickly created, which is useful in designing robots with parallel mechanisms. The laminates are built from layers of cardboard or fiberglass to add rigidity. These outer layers are placed on both sides of a thin polyester film or nylon fabric to create flexible joints. Sheets of thermo-set adhesives bond each layer together after being heated in a heat press. Originally, patterns were defined for cutting using Python to generate early prototypes of the device; SolidWorks and popupCAD [5] were used in later iterations. Each layer of material is cut out individually on a laser cutter, and then stacked and bonded together. The resulting laminate is cut once again, releasing it from the surrounding web of scrap material. The laminates created by this process are then assembled into a robot by connecting actuators, plastic rivets, and metal screws to a 3D-printed frame. Because parts can be quickly disassembled at the connection points, this process allows us to rapidly prototype and test a variety of leg transmissions in a short period of time. Fiberglass is used as the rigid layer in the final iteration because of its stiffness, durability, and ability to be cut with a laser cutter.

6 Mechatronics Design

The current quadruped robot uses eight Dynamixel XL-320 servos to control the two degrees of freedom across four legs. The Dynamixel servos are daisy-chained in two pairs of four control lines, with each powered by a 7.4V 1300mAh lithium polymer battery in parallel. The OpenCM 9.04 microcontroller is connected to the first Dynamixel in the sequence and is able to send commands to and read parameters from each servo. The OpenCM 9.04 is programmed using the Arduino IDE, using packages published by its manufacturer, Robotis, to integrate it into the Arduino IDE. This allows the microcontroller to be programmed using a widespread and well-known IDE, and allows other packages and devices to integrate with it easier. To measure the robot's motion we have selected the Adafruit BNO055 IMU over I2C using a modified version of the existing Adafruit

Table 2. Equipment and Materials List

Equipment	Materials
Laser Cutter	Cardboard
Heated Press or Iron	Nylon Fabric
CAD Software	0.2” Carbon Fiber

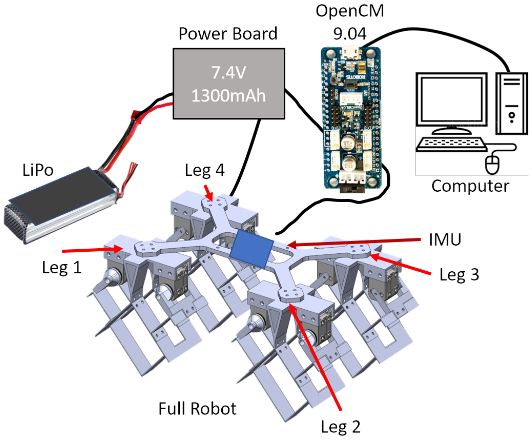

Fig. 5. Diagram of CAD quadruped with electrical components attached

BNO055 library. Each discrete component is placed in order to evenly distribute the weight of the robot as much as possible and keep the center of mass at the two symmetric planes of the robot. The IMU is placed at the center of the robot so that it can properly send orientation data and respond to those changes. A figure of the robot’s layout can be seen in Figure 5. The equipment and materials required to create the robot are seen in Table 2.

7 Evaluation

An assortment of tests were performed on the robot to compare the motion of the prototype to the motion simulated in Python. The following three tests were performed using equations (1) and (2) as servo inputs and can be seen in Figure 6.

Test 1: In Figure 6a, the robot can move up and down by programming each servo with a 180°difference between ϕ_{i1} and ϕ_{i2} for the two servos in the leg. The simulated output from this trajectory predicts the foot can lift itself about 7.5 cm along the Y-axis. On the prototype it is able to lift its leg around 7 cm when going through the same overall motion trajectory.

Test 2: A similar comparison can be made in Figure 6b, when the robot is leaning forward and backward; by controlling the servos with a difference of 0°applied to the motor control parameters ϕ_{i1} and ϕ_{i2}. In simulation the robot is able to move 7.4 cm, while in application it can move around 4.5 cm.

Test 3: Figure 7 shows the trajectory when a 90°difference is applied to servo parameters ϕ_{11} and ϕ_{12} on leg 1. This creates a cyclic motion at the foot that is similar to the simulated end-effector trajectory. In the simulation, the robot is able to move 5.5 cm up and down, while in application the robot is able to move around 5 cm. The simulated robot is also able to move 5.6 cm back and forth, while in application the robot moves around 5 cm.

The ability of the robot to move itself was evaluated both when the robot was unloaded, and when it was loaded with varying weight. We used a bag of rice to allow us to vary weight incrementally; it was placed on top of the robot close to the robot’s center of mass. The robot was then controlled as described previously.

The robot weighs 618 g, and is able to lift itself up linearly (Test 1) at a payload of 100 g consistently; it could pick up a 200g payload half the time; when it failed the motors faulted due to an over-current condition; it failed to lift the 300 g payload. The robot was able to move itself forwards and backwards (Test 2) up to around 200 g and 300 g easily, but struggled at 400 g.

8 Discussion

Based on the evaluation, we can discern that the robot is consistently able to get within 1 cm of the expected target distance from the Python model in Test 1 and Test 3, and struggled when completing Test 2. The reason that it is unable to match the predicted motion more closely is due to the stiffness of the laminate joints; as they deflect more than anticipated. The leg is also limited by mechanical interference at the servos’ joints, making it difficult to match the range of motion predicted by the Python simulation. Near the extremes of the actuators’ ranges of motion the mechanism is also closer to a singularity, making it difficult for the robot to lift itself with such a low gear ratio. Another possible reason the robot had difficulty lifting itself may be due to power and current limitations of the selected battery.

The robot is able to hold itself up on three legs, and can also move another leg in a sine wave without making it fall over, meaning that it can be coded with a brute force walking gait as well as a more complex control method. It can also lift a 100 g payload vertically, meaning it can lift about a sixth of its 618 g body weight. Its capabilities and low cost also make it suitable for many classroom projects with applications including controls, modeling, machine learning, and design optimization.

This platform already has a strong track record in the classroom. It was initially developed as a project for *Foldable Robotics*, a graduate-level class taught at Arizona State University in 2016. Since then it has been iteratively modified over several years in order to improve its reliability, stiffness, and payload. The platform was redesigned in Fall 2018 for a controls design project; in that project, each leg was

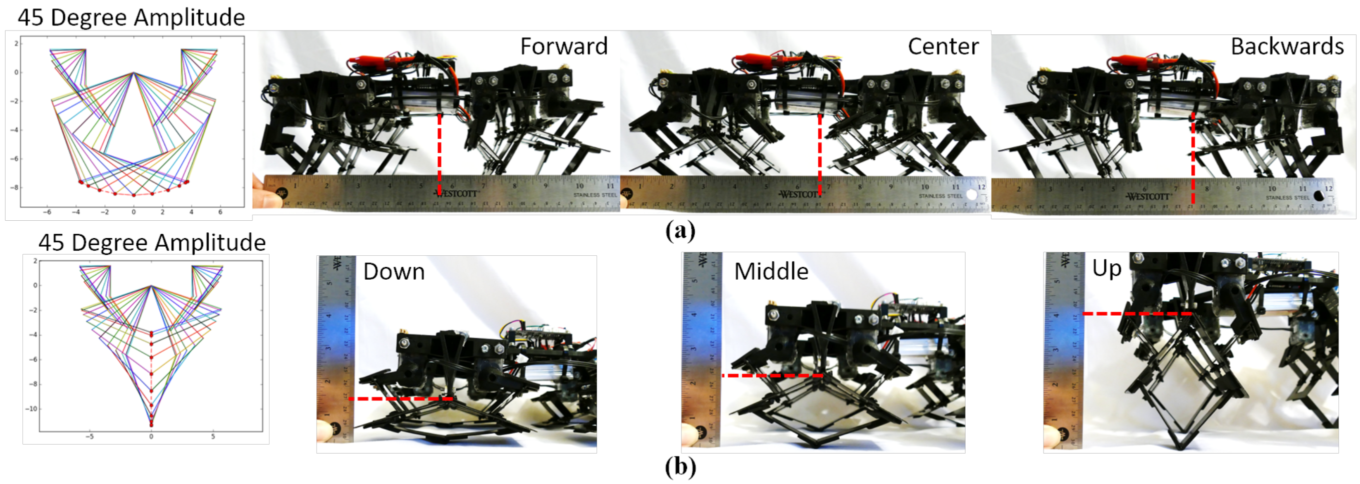

Fig. 6. (a) Python kinematics simulation for servos moving with a 45° amplitude by a 180° difference between ϕ_{i1} and ϕ_{i2}, programmed into the robot causing it to lift up and down. (b) Python kinematics simulation for servos moving with a 45° amplitude by a 0° difference between ϕ_{i1} and ϕ_{i2}, programmed into the robot causing it to move back and forth

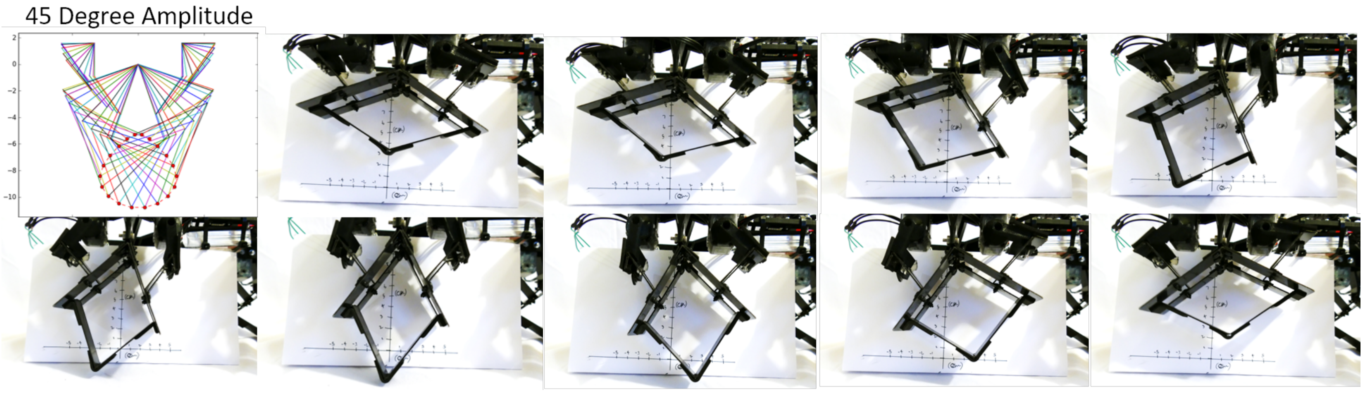

Fig. 7. Python kinematics simulation for servos moving in 90° phase offset with pictures of actual robot leg moving throughout the trajectory using the angle range as what is programmed in the simulation

treated as a single-input-single-output system controlled by a PID controller; this robot was then designed to balance itself based on sensor outputs from the servos and the integrated IMU. In the 2019-2020 academic year, students are continuing to work with this platform by adding force sensors to the feet along with a tail in order to determine how these elements may affect and improve the platform's performance in balancing, walking, and eventually running. The data that can be gathered from the on-board servos and IMU sensors will make it ideal for future machine learning and neural network applications as well.

To make this robot accessible to classrooms and research labs, we have shared a parts list, a short program written for the Arduino IDE that moves the servos, CAD design files, a Python script for analyzing the kinematics of the system, and an instructional guide on how to assemble the platform on our website [19]. By using the CAD files and following the instructional guide, the platform can be replicated; the programs supplied can be used as a starting point to have the robot move its servos, and altered to make the robot perform in a desired manner. The kinematics files can also be altered to simulate various leg designs build on the current model. These tools provided will not only make assembly of the robot intuitive, but also give students the chance to learn and incorporate their own ideas into the design.

9 Conclusions and Future Work

In this paper we have presented the design and application of a two degree-of-freedom laminate leg transmission. The platform can be used in robotic research applications to solve locomotive control problems; it is a robust, accessible, and easily modifiable robotic platform. This platform can be built using simple manufacturing methods and low-cost materials. It was designed through a combination of kinematic design tools and CAD modeling. The system underwent a series of design changes supported by several prototypes, eventually leading to the implementation of laminate parallel mechanisms that serve as the basis of the current leg design. Using two actuators per leg creates a rich control space, permitting the study of gaits, balance, high-level control, and task-based decision making. This transmission provides more functionality than robots of similar scales – its control problems are only partially solved through careful mechanical design and analysis.

This device would benefit from being recorded with motion capture equipment so that the specific distance measurements taken at the end-effector can be more accurately analyzed and compared to the simulation. The Python simulation would benefit by including the torque and force calculations necessary to understand our motors' torque limitations, something that is not discussed in this paper. This platform would also benefit from more accurate models that take into account flexible link dynamics. Related work surrounding the development of a laminate hopping robot has been recently presented[20]; our goal is to unify these two platforms using the modeling and performance data collected for each system into a next-generation platform with higher performance.

Due to modeling errors imposed by soft, compliant links, it is difficult to apply traditional theoretical controls concepts to less-expensive systems such as the platform presented above. Thus, this platform motivates several new avenues for study both in research and in the classroom. Motivated by the potential to use lower-cost makes robots such as these will require engineers to learn and develop optimal control methods tuned for practical, compliant, and low-cost systems with higher degrees of modeling uncertainty. This platform serves as a great tool for students with the desire to create more descriptive models that consider system compliance; through the shared design and control files they will be able to easily replicate this system and test compliance-aware dynamic models to see what effects these elements have on system performance.

The quadruped presented in this paper offers a specific application of the transmission in use. This robot will allow for the research of quadrupedal walking control, a topic that can be difficult to research without a robust platform. Many of the existing platforms for this research are expensive, and laminates offer an inexpensive yet robust platform. It is able to move itself using a brute force programming approach and has the potential to be controlled by more complex closed-loop controls. It also offers a design methodology that uses the synergistic combination of accessible simulation, rapid prototyping, and a modular approach making it possible to quickly adapt this design for specific uses. Research labs, schools, and other academic institutions can create their own version of this quadruped with common manufacturing tools they already have, and for a lower cost than similarly capable platforms. The quadruped will also be a topic of continuing robotics and controls research in the IDEAlab.

References

[1] Gollnick, P. S., Magleby, S. P., and Howell, L. L., 2011. "An Introduction to Multilayer Lamina Emergent Mechanisms". *Journal of Mechanical Design,* **133**(8), p. 081006.

[2] Aukes, D. M., Ozcan, Ö., and Wood, R. J., 2014. "Monolithic Design and Fabrication of a 2-DOF Bio-Inspired Leg Transmission". In *Conference on Biomimetic and Biohybrid Systems*. Springer International Publishing, Milan, pp. 1–10.

[3] Aukes, D. M., Goldberg, B., Cutkosky, M. R., and Wood, R. J., 2014. "An analytic framework for developing inherently-manufacturable pop-up laminate devices". *Smart Materials and Structures,* **23**(9), sep, p. 094013.

[4] Aukes, D. M., and Wood, R. J., 2014. "Algorithms for Rapid Development of Inherently-Manufacturable Laminate Devices". In Volume 1: Development and Characterization of Multifunctional Materials; Modeling, Simulation and Control of Adaptive Systems; Structural Health Monitoring; Keynote Presentation, Vol. 1, ASME, p. V001T01A005.

[5] Doshi, N., Goldberg, B., Sahai, R., Jafferis, N., Aukes, D., Wood, R. J., and Paulson, J. A., 2015. "Model driven design for flexure-based Microrobots". In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, pp. 4119–4126.

[6] Khodambashi, R., Sharifzadeh, M., and Aukes, D. M., 2018. "An Integrated Design and Simulation Environment for Rapid Prototyping of Laminate Robotic Mechanisms". In Volume 5B: 42nd Mechanisms and Robotics Conference, American Society of Mechanical Engineers, p. V05BT07A068.

[7] Baisch, A. T., Ozcan, O., Goldberg, B., Ithier, D., and Wood, R. J., 2014. "High speed locomotion for a quadrupedal microrobot". *The International Journal of Robotics Research*, may.

[8] Goldberg, B., Doshi, N., Jayaram, K., and Wood, R., 2017. "Gait studies for a quadrupedal microrobot reveal contrasting running templates in two frequency regimes". *Bioinspiration & Biomimetics,* **12**(January).

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

[10] Hoover, A. M., Burden, S., Fu, X. Y., Sastry, S. S., and Fearing, R. S., 2010. "Bio-inspired design and dynamic maneuverability of a minimally actuated six-legged robot". *2010 3rd IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob 2010*(May), pp. 869–876.

[11] Mulgaonkar, Y., Araki, B., Koh, J. S., Guerrero-Bonilla, L., Aukes, D. M., Makineni, A., Tolley, M. T., Rus, D., Wood, R. J., and Kumar, V., 2016. "The flying monkey: A mesoscale robot that can run, fly, and grasp". *Proceedings - IEEE International Conference on Robotics and Automation,* **2016-June**, pp. 4672–4679.

[12] Wang, L., Yang, Y., Correa, G., Karydis, K., and Fearing, R. S., 2019. "OpenRoACH: A Durable Open-Source Hexapedal Platform with Onboard Robot Operating System (ROS)". In 2019 International Conference

on Robotics and Automation (ICRA), IEEE, pp. 9466–9472.

[13] Luck, K. S., Campbell, J., Jansen, M., Aukes, D. M., Ben Amor, H., Amor, H. B., and Ben Amor, H., 2017. "From the Lab to the Desert: Fast Prototyping and Learning of Robot Locomotion". In Proceedings of Robotics: Science and Systems.

[14] Jansen, A., Luck, K. S., Campbell, J., Amor, H. B., and Aukes, D. M., 2017. "Bio-inspired Robot Design Considering Load-Bearing and Kinematic Ontogeny of Chelonioidea Sea Turtles". In Conference on Biomimetic and Biohybrid Systems, Springer, pp. 216–229.

[15] Raibert, M., Blankespoor, K., Nelson, G., and Playter, R., 2008. "BigDog, the Rough-Terrain Quadruped Robot". *IFAC Proceedings Volumes,* **41**(2), pp. 10822–10825.

[16] Remy, C. D., Siegwart, R., Gehring, C., Bloesch, M., Hoepflinger, M. a., and Hutter, M., 2012. "A Compliant Quadrupedal Robot for Fast, Efficient, and Versatile Locomotion". *15th International Conference on Climbing and Walking Robot - CLAWAR 2012*, pp. 1–8.

[17] Spröwitz, A., Tuleu, A., Vespignani, M., Ajallooeian, M., Badri, E., and Ijspeert, A. J., 2013. "Towards dynamic trot gait locomotion: Design, control, and experiments with Cheetah-cub, a compliant quadruped robot". *International Journal of Robotics Research,* **32**(8), pp. 932–950.

[18] Guzzio, E., 2010. Boston Dynamics Wins Darpa Contract To Develop LS3 Robot Mule (It's a Bigger BigDog) - IEEE Spectrum.

[19] Aukes, D. M., 2019. "IDEAlab Download Page". *http://idealab.asu.edu/download.*

[20] Knaup, J. W., and Aukes, D. M., 2019. "IDETC2019-98109".