DER Simulation of the World's Highest Jumping Robot Final Report

Junlin Chen[1] and Sanskar Nalkande[2]

Abstract—**This paper presents a numerical simulation method for jumping robots. Starting with a literature review of existing jumping robots, this paper then introduces the hybrid tension compression jumping robot, which is the subject of the simulation. A significant portion of the study is dedicated to the hybrid tension compression robot, noted for its exceptional ability to leap 31 meters high, achieved through a novel hybrid tension-compression structure. The paper delves into the discrete elastic rods (DER) theory for simulation, employing the DisMech simulation tool for soft robots and structures. The simulation work includes detailed modeling decisions like material selection, structural updates, and simplifications to facilitate comparisons with real-world performance. Results indicate a quasi-linear relationship exists between the launch velocity and the compression ratio of the simulated robot, proving a successful simulation of the simplified robot. Although the simplified model exhibits lower performance compared to the real experiment, the discrepancy is explained in the conclusion section of this paper.**

Index Terms—**Numerical Simulation, Tension-Compression Structure, Jumping Robot, DER.**

I. INTRODUCTION

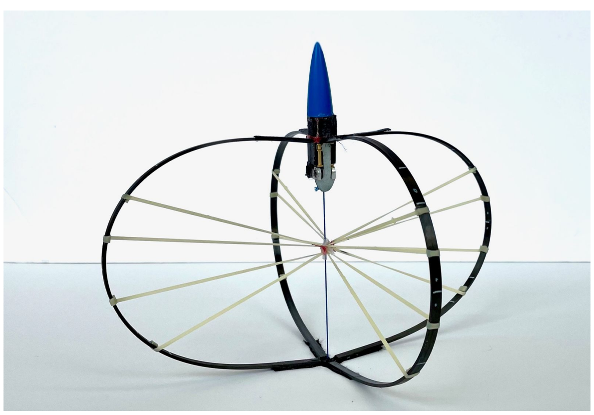

Fig. 1. The world highest jumping robot from UCSB[2]

There are various reasons why a robot needs jumping capability. For two-legged robots or track-based/wheel-based robots, their ability to cross rough terrains is limited by their physical parameters (i.e. track length, leg height) and ground surface properties. On the other hand, a jumping robot of any size may overcome the roughest terrain by simply ”flying” over it. In low-gravity environments, such as the moon, jumping robots can leap hundreds of meters in height. A jump at this height, if can be directed precisely, have the potential to be utilized as a means of movement to overcome craters and canyon on extraterrestrial surfaces.

Furthermore, given the main driving components are just elastic rods and electric motors, jumping robots can be made lighter than traditional robots with legs or tracks. As a result, this type of robot will be particularly valuable in extraterrestrial exploration and space missions for its lightness and ability to go across terrains that are unimaginable by existing robots or rovers.[10]

However, not all jumping robot designs are created equal. The jumping height of a robot is highly dependent on the elastic structure it uses.

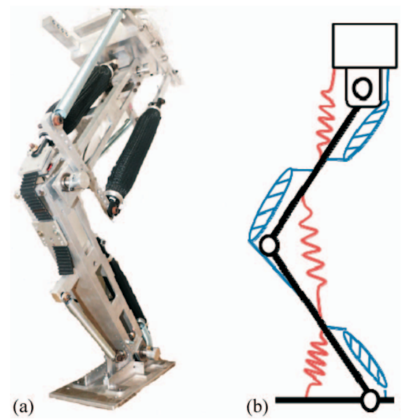

Fig. 2. The one-legged robot[8]

1) One-legged robot: The one-legged jumping robots are generally biomimicry of a real leg, with springs and electric motors instead of tendons and muscles. The spring and motor combination provides the robot with the ability to stretch and store elastic energy. As hinged motors gradually unwind to decrease the angle between rods, the springs will be stretched for elastic energy in springs to increase. When rods arrive at the designated orientation, the spring tension will be released while motors inside the joints quickly apply torque to send the robot into the air. All of this is explained in detail in a paper by D. Tian et al.[6] Unfortunately, due to limitations on elastic material and the shear weight of the whole structure, one-legged robots generally perform poorly in terms of jumping height when compared to other designs.

2) Three-legged robot: The major difference between a three-legged structure and an oner-legged one is the way of storing energy inside the structure. Three-legged robots mainly utilize tension linkage, which is a structure with freely movable limbs and elastic bands in the middle as the means of energy storage.

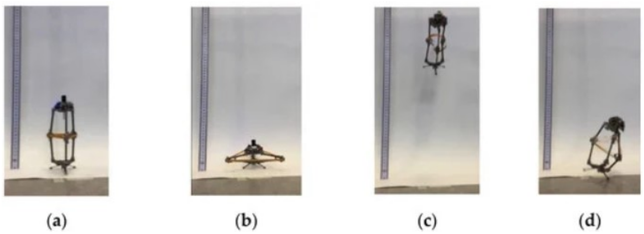

Fig. 3. The three-legged robot[7]

In the paper by Y. Yan et al.[7], they demonstrated a three-legged robot with four limbs and four elastic bands. The four limbs are interconnected via hinges, which allows the limbs to stretch elastic bands as the structure opens up. In Yan's paper, they managed to obtain a maximum jump height of 4.3718 meters[7], proving the effectiveness of tension linkage.

3) UCSB robot: Recall that this project focuses on the world's highest jumping robot that can soar 31 meters above the ground with a body length of merely 0.3 meters. To achieve a jump of this height, this robot utilized an engineered drive system along with the novel hybrid tension-compression structure to best improve the specific elastic energy stored by the system.[1] In his paper, Hawkes proved the new structure is able to obtain higher specific elastic energy under the same amount compression rate of 0.2. The use of the new elastic structure and the success of the robot incentivize the digital simulation of such robots for future parametric study and preliminary design. If proven successful, the simulation method implemented in this project can act as a foundation stone for future research on high-performing jumping robots or hybrid tension-compression structures.

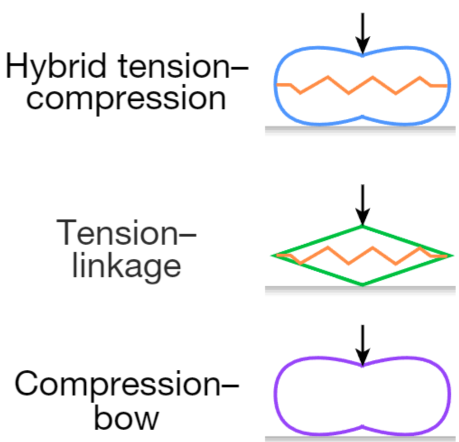

Fig. 4. Different compression structure layout from Hawkes et al.[1]

II. SIMULATION BACKBONE

A. Discrete Elastic Rods

The backbone of simulation is the DER theory introduced by Mikl Bergou et al. in 2010[9]. For DER, each rod is divided into several edges and vertices, and the states of these edges and vertices are predicted based on the equation of motion at each time step.

$$M\ddot{q} + \frac{dE_{elastic}}{dq} = f_{external} \tag{1}$$

where

1) $M = massofthesystem$
2) $q = thepositionofthesystem$
3) $E_{elastic} = theelasticenergyofthesystem$
4) $f_{external} = externalforceexperiencedbythesystem$

The equation of motion is solved numerically at each time step and each edge or node to obtain the respective position of nodes and edges. The numerical method utilized in the project is the Newtonian method. At each iteration, the predicted solution(x_{n+1}) is updated using the following formula:[3]

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{2}$$

The numerical process continues to reduce the error until the error meets the corresponding threshold. To facilitate the Newtonian method calculation, the equation of motion can be rearranged and discretized to:

$$0 = \frac{m_i}{\Delta t}\left(\frac{q_i(t_{k+1}) - q_i(t_k))}{\Delta t} - \dot{q}_i(t_k)\right) + \frac{dE_{elastic}}{dq_i} - F_{i,external} \tag{3}$$

By using the Newtonian method to solve for the $q_i(t_{k+1})$ either implicitly or explicitly, the system can be simulated throughout time.

B. DisMech Simulation

DisMech is a physical simulation built specifically for soft robots and structures. Thanks to the members of the Structures-Computer Interaction Lab at UCLA, the simulation can be done on a user-friendly interface. DisMech offers functions that allow users to easily create geometry through limb() and joint() commands. DisMech also carries other functions that enforce force application, edge fixture, data logging, and other packages that facilitate the simulation of rod-based structures. All commands and parameters are specified inside the robotDescription.cpp file and an OpenGL-based graphical representation of the simulation will automatically pop up for easy visualization. Nevertheless, to better realize the goals of the project, four major modifications are made to Dismech. Among these, three of the changes are located inside the world class. First of all, two more public double variables are added to the class to keep track of the maximum compression height and maximum velocity during the simulation. The updateTimeStep() function is then modified so that after a certain time step, the uniform force applied at the beginning of the simulation will be erased for the robot to launch. Lastly, the printSimData() function is modified. When the ground contact condition is added, the printSimData function will now print the current velocity and compression rate of the robot in addition to all other parameters.

III. MODELING DECISIONS

A. Material Selection

After a deep dive into the UCSB paper, the author found that the carbon fiber bows used by the UCSB team come from ACP Composites. The exact properties of the carbon fiber they used are obtained from the specification sheet published by ACP Composite.[2]

TABLE I
ACP COMPOSITES CARBON FIBER PROPERTIES

Tensile Strength	1.72 GPa
Shear Modulus	5 GPa
Poisson Ratio	0.1
Density	$1.2g/cm^3$
Ultimate Tensile Strain	1.50%
Composite Type	0° unidirectional orientation carbon fiber

The above properties will be applied to the DisMech simulation for each carbon fiber bow and the crosses.

B. Structural Updates

Initially, the hybrid tension-compression structure is constructed on DisMech:

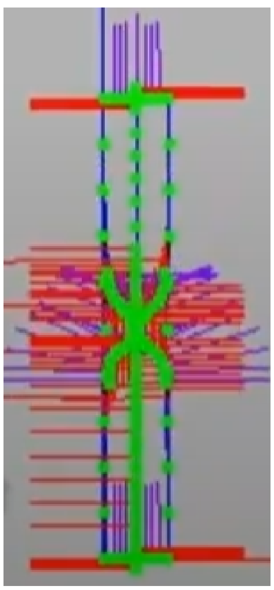

Fig. 5. Current structure of the robot at the natural condition

The four bows are connected to two crosses at the system's top, which forms a compression bow structure. When force is acting on the top of the system, the bows will compress and store elastic energy. The tension linkage system is then added by connecting one end of the eight elastic bands to a central rod fixed at the middle of the bottom cross and the other end to the carbon fiber bows. The elastic bands will have drastically different material properties compared to carbon fiber. Thanks to the interface of DisMech, such problems can be addressed easily by setting different parameters when constructing the corresponding limbs.

C. Simulation Progress

During simulation, the initial and boundary condition of the robot is defined. The robot will have no initial displacement and velocity while having uniform force applied constantly at the top cross. Each and every node of the robot is free to move, but a soft ground contact is enforced at the bottom of the robot with a stiffness of 100000. By combining ground contact and uniform force, the robot will deform in a way that matches its real-world counterpart. The following image shows the compressed shape of the robot.

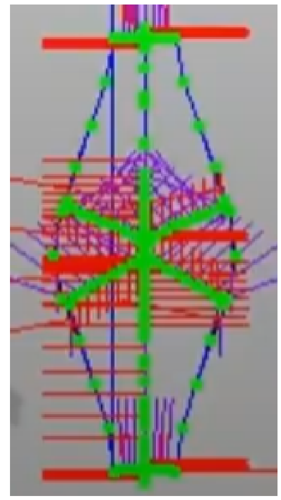

Fig. 6. Structure of the robot at the compressed condition

The time step solver used in the simulation will be the backward Euler method with a time step of 10^{-4} seconds. The backward Euler integration method can be visually understood by the figure below.[5]

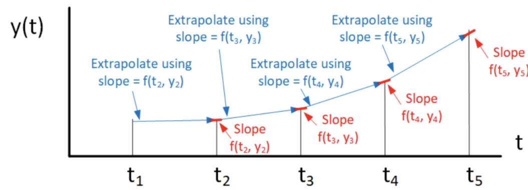

Fig. 7. Visualisation of Backward Euler Integration Method

D. Model Simplification

Even though a hybrid tension compression model can be constructed and simulated inside DisMech, the simulation is erroneous. During the simulation process, the total amount of deformation for the hybrid structure needs to be strictly regulated to prevent non-convergence. The maximum amount of energy input(calculated by work=Force * distance) is limited to 11 joules during the simulation. Any higher input will result in non-convergence during the launching process. On the other hand, if the tension linkage structure is removed from the system, the simulation will become robust. Hence, to make a meaningful comparison between the simulation and the real-world result, the hybrid system is reduced to a compression bow. Below are the images for a compression bow structure at uncompressed and full compression state:

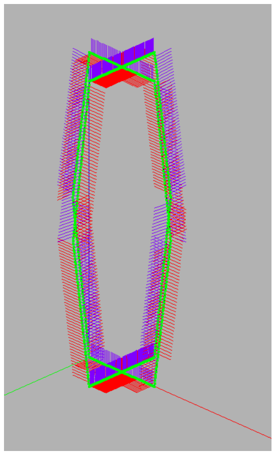

Fig. 8. Uncompressed compression bow model

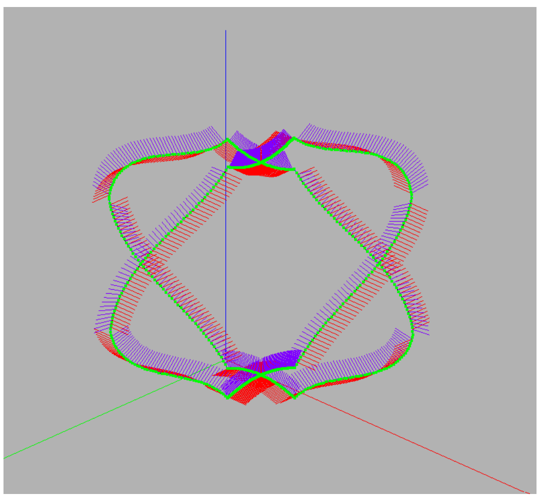

Fig. 9. Fully compressed compression bow model

IV. Results

A. Compression Simulation

The simplified model is able to achieve a similar compression pattern compared to the real robot.

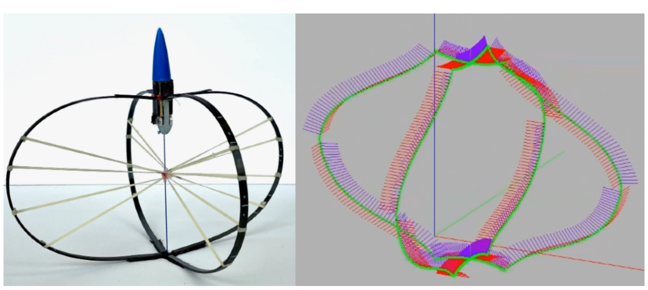

Fig. 10. Comparison of Compressed structures in real life and in simulation

B. Jumping Simulation

The simplified model is able to achieve a similar release and jump pattern compared to the real robot.

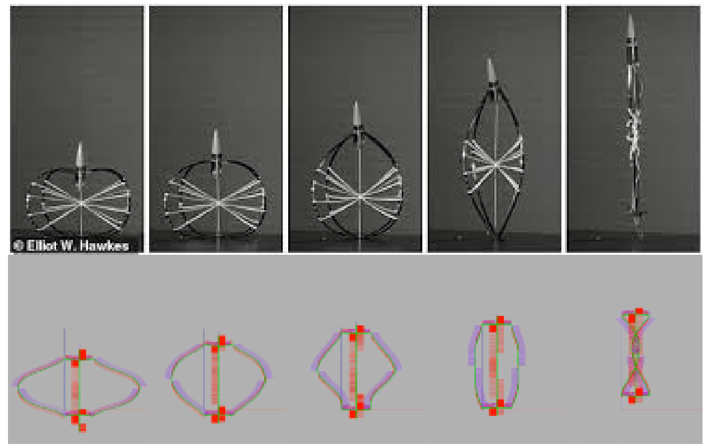

Fig. 11. Comparison of Compressed structures in real life and in simulation

C. Compression Ratio

To compare measurable metrics such as jumping height and velocity, one cannot use the actual compression strain of the rods due to the lack of measurement tools. E. W. Hawkes et al.'s paper introduces a quantity called the compression ratio, which is the ratio between the height change during compression and the original height[1]. The formula of compression ratio is defined as:

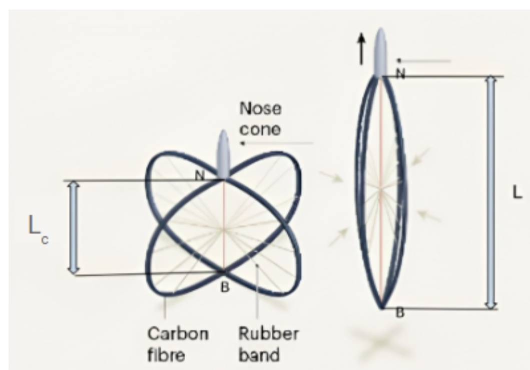

Fig. 12. Compression ratio

$$\text{Compression Ratio} = \frac{L - L_c}{L} \tag{4}$$

The compression ratio can be changed by applying more force on the robot or applying the same amount of force longer. This characteristic of compression ratio means it has an underlying connection to the amount of energy put into the system.

D. Effects of Compression ratio

Now for each of these cases where the compression ratio is different, the jumping velocity and maximum height of the robot are recorded. The results are represented in the following plots.

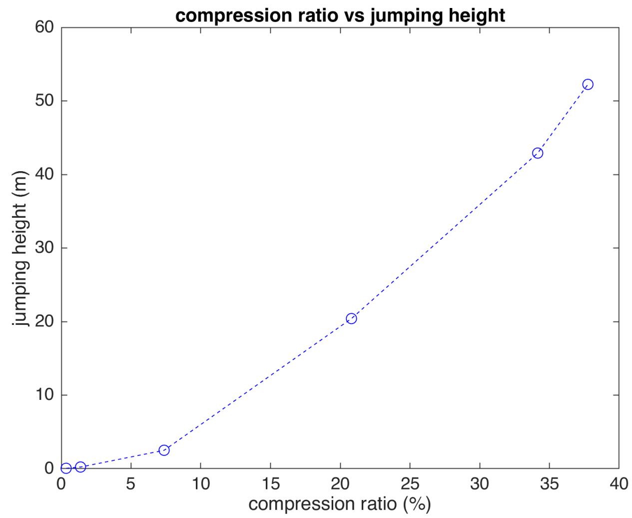

Fig. 13. Maximum Height reached (in meters) vs compression ratio(%)

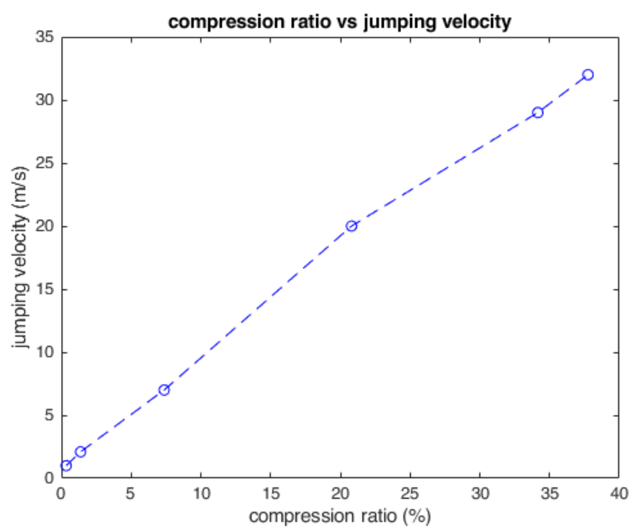

Fig. 14. Launch Velocity (m/s) vs compression ratio (%)

According to the graphs above, there is a quasi-linear relationship between the launch velocity and the compression ratio, matching the real-world characteristics of a compression bow system.

V. Conclusion

For a meaningful comparison between the simulation and the real-world result, the geometry and material properties of the robot come directly from the paper published by E. W. Hawkes et al. For the parameters given in the paper, the real-life robot attained a maximum jumping height of 31 meters at a compression ratio of 20%. In terms of the simulated robot, the robot achieved a maximum jumping height of 21 meters at a compression ratio of 20%, which is lower than the real experiments. The possible reasons for this lack of performance are listed below:

- The lack of tension linkage means less elastic energy is stored during the same compression process.
- The real-world compression is a long process that lasts 90 seconds. However, to save simulation time, the compression process in DisMech only took 12ms. Due to the much faster compression, inertia will keep compressing the structure even after the force is lifted. This means under the same compression ratio, the energy input of a faster process is lower than the energy input of a slower process.

Acknowledgment

The authors would like to thank Andrew Choi for his support when dissecting the DisMech source code. The authors would also like to thank Professor M. Khalid Jawed for offering M263F during Fall 2023. M263F provided the authors with fundamental knowledge of DER, which is the basis of this project.

References

[1] E. W. Hawkes et al., ”Engineered Jumpers Overcome Biological Limits via Work Multiplication,” Nature News, Apr. 27, 2022.

[2] ACP Composites, ”Solid Carbon Rod All Series PDS,” No.49881, Oct.2019.

[3] M. Khalid Jawed, Class Lecture, Topic: "Networks of Beams." Module 17, University of California, Los Angeles. Oct, 2023.

[4] W. Simchareon, T. Amnuaikit, P. Boonme, W. Taweepreda, and W. Pichayakorn, ”Characterization of Natural Rubber Latex Film Containing Various Enhancers,” Procedia Chemistry, vol. 4, pp. 308-312, 2012, ISSN 1876-6196. DOI: 10.1016/j.proche.2012.06.043.

[5] “1.3: Backward Euler method,” Mathematics LibreTexts, Jul. 17, 2022. https://math.libretexts.org/Bookshelves

[6] D. Tian, J. Gao, C. Liu, and X. Shi, ”Simulation of Upward Jump Control for One-Legged Robot Based on QP Optimization,” Sensors, vol. 21, no. 5, p. 1893, Mar. 8, 2021.

[7] Y. Yan, K. Smith, A. Macario-Rojas, and H. Zhang, ”Simulation of the Landing Buffer of a Three-Legged Jumping Robot,” Machines, vol. 10, no. 299, 2022.

[8] I.-F. Lee, C.-I. Lin, H.-C. Wu, L.-C. Wu, P.-C. Lin, M.-H. Chiang, and W.-P. Shih, ”Squat and Standing Motion of a Single Robotic Leg Using Pneumatic Artificial Muscles,” Journal of Applied Science and Engineering, vol. 18, no. 4, Dec. 2015.

[9] Miklós Bergou, Basile Audoly, Etienne Vouga, Max Wardetzky, and Eitan Grinspun., ”Discrete viscous threads”, ACM Trans. Graph. 29, 4, Article 116, 2010. https://doi.org/10.1145/1778765.1778853

[10] O’Neill, I. J., Skelly, C. (2020, June 24). This Hopping Robot Could Explore the Solar System’s Icy Moons. NASA. https://www.nasa.gov/solar-system/this-hopping-robot-could-explore-the-solar-systems-icy-moons/