Trebuchet

Dylan Tamras
Illinois Institute of Technology
Undergraduate

Lillian Imley
Illinois Institute of Technology
Undergraduate

Rawdah Abdullah
Illinois Institute of Technology
Undergraduate

Ayler Orasheva
Illinois Institute of Technology
Undergraduate

Abstract- Trebuchet uses a counterweight to store energy to launch a projectile. The range of the trebuchet relies on the counterweight's mass and the lever arm. This paper will examine the effect of the trebuchet's arm length on its range.

I. Introduction

The counterweight trebuchet was a medieval siege engine first used in the 13th century. Its advantage over the catapult was that it didn't require a tightly strung cord and rope. The counterweight trebuchet works by having a sling attached to one end of an arm and a counterweight attached to the other end. When the counterweight is released, the sling end of the arm will lift, and the projectile will be launched and released (Chevedden, 2000).

We were tasked with creating our own counterweight trebuchet that launched a distance of 6 meters and used an Arduino connected to a servo to release the counterweight.

II. THEORY

As previously stated, when the counterweight of the trebuchet is released, the opposite side of the arm where the projectile gets launched. In more technical terms, the trebuchet converts potential energy into kinetic energy to launch said projectile. The potential energy is stored in the counterweight. As it falls towards the ground, the potential energy converts into kinetic energy.

As the counterweight hits the ground, the arm builds up angular velocity, causing the projectile to accelerate outwardly; most of the conversion from potential to kinetic energy occurs in this state. When the projectile is finally launched, energy conversion from potential to kinetic energy has been largely converted. By the time the launch is completed, most of the energy has been released, leaving the arm swinging with the leftover energy.

III. APPARATUS

Through MATLAB code, the measurements for the trebuchet were determined. Using those parameters, the prototypes were developed using the Autodesk Inventor CAD software. From the two prototypes, the optimized design was chosen. The focus was on making sure there was space for the counterweight to fall freely and for the arm to launch from the right distance at the right angle.

The design of the trebuchet adheres to conventional principles, as illustrated in Fig. 1 (Spenko, 2019). Two triangular trusses are positioned on either side of a runway piece that lies between the two trusses. The two trusses feature three cutouts along their base, each spaced 100 mm apart, allowing the runway strip's extruded pieces to be pushed into place. Wood glue was applied at these connection points to strengthen the trebuchet further.

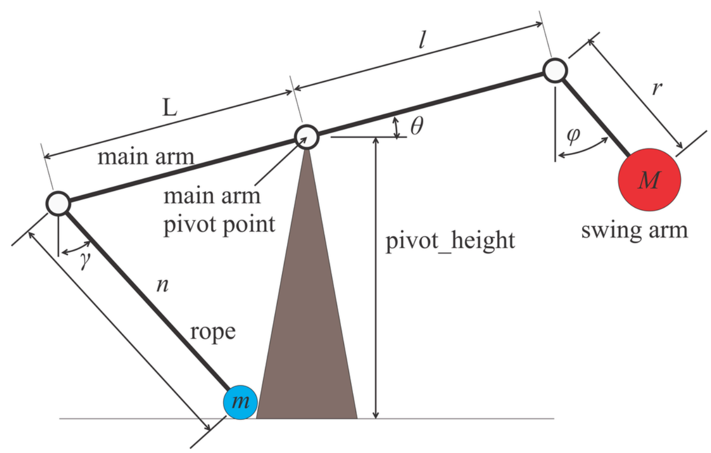

Fig. 1 Illustration of a conventional trebuchet design that inspired our design, where the triangular piece in the middle acts as the truss, pinning the throw arm in place and allowing for the sling to be at one end, while the counterweight resides at the opposite end.

Additionally, on the outer side of the two trusses is another triangle that is connected similarly to the runway strip. This extra triangle was also reinforced with wood glue in its connection slots. These triangles connected perpendicularly to the truss and sat flush on the ground to support the main trusses and keep the entire structure upright (see Fig. 2).

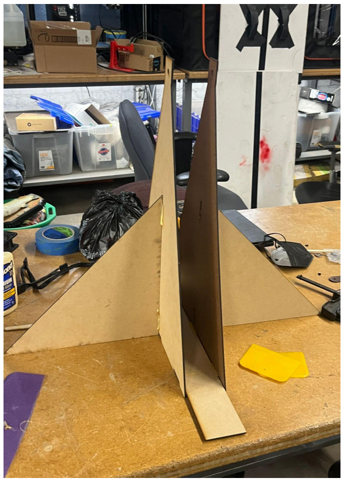

Fig. 2 Basic design including the two trusses on either side of the runway strip, followed by an additional two triangular supports to keep the entire structure upright.

Next, a ¼ inch hole was cut out at the top of each truss to allow for the acrylic rod to slide between the two trusses, creating a pivot point for the throw arm to revolve around. The throw arm itself was 780 mm long, with one end holding the trebuchet's counterweights and the other anchoring the sling.

After fastening the arm between the two trusses, the side with the counterweights is leveraged higher than the arm holding the sling. The sling, a disposable surgical mask, is then pulled along the runway strip between the two trusses until the nylon strings are held in tension. Then, an extra length of nylon is pulled past the sling and clamped down by a servo. Upon sending a signal from a laptop to the Arduino, causing the clamp to rotate upwards, releasing the extra piece of nylon and allowing the counterweight of the arm to fall, pulling the sling through the trusses and up, releasing the small rubber ball that was loaded as ammunition.

The design for the trigger (see Fig. 3) was also determined. The trigger needed to be detachable and work with a 350 Nmm servo. For that reason, a simple, straight, thin piece of MDF board was cut out and attached using a servo horn. Then, code to turn the trigger 180 degrees was developed.

IV. EXPERIMENTAL SETUP

First, the side of the arm holding the counterweight was lifted up so that the sling could rest upon the runway strip between the two truss supports. The sling was then pulled through the space between the two trusses until it sat in tension.

Next, a 27 g rubber ball was loaded into the sling. An additional string that was attached to the same anchor point as the nylon strings holding the sling was also pulled through until it reached the opposite side of the arm (below the counterweights). This string was also pulled to tension and secured using the trigger mentioned before. This trigger clamped the nylon string down and held the entire trebuchet in equilibrium until a signal ran through the servo lifted the arm.

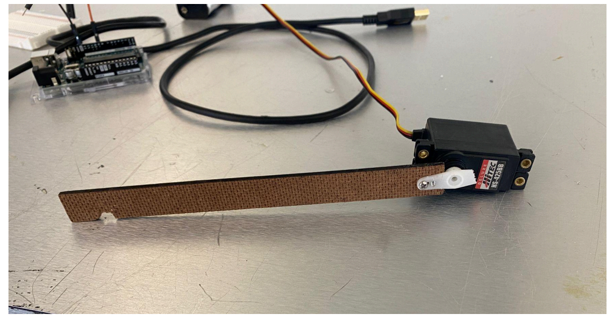

Fig 3. The trigger design used to clamp down the length of the nylon that, when released, would allow for the throw arm to swing around and fire its ammunition. The small cut-out at the end would act as a hook to keep the nylon in place until ready for launch.

At this point, all that is needed to launch the trebuchet is to send a signal to the servo, raising the trigger and releasing the string it held down, allowing for the counterweights to drop and the arm to swing.

V. EXPERIMENTAL PROCEDURE

The goal of this project was to successfully launch the 27 g rubber ball a minimum distance of 6 meters. Our initial trials failed to meet this criterion, and improvements to our design had to be made.

We began by trying different lengths of the nylon string that held the sling. At first, we used a length that was far too short. This made it difficult to pull the sling through the space between the two trusses, thus resulting in a poor launch distance. To solve this problem, we increased the length of the nylon string until we could pull the entire length between the two trusses and hold it in tension. This change allowed for much better launch distances and brought us closer to our target of 6 meters.

The other significant change we made also had to do with the area containing our sling. Initially, we discovered that the hook piece holding the nylon string in place was too short. We ran several trials where we increased the length of the hook to find the optimal length that would result in the sling becoming unhooked at the right moment, causing a greater launch distance. Eventually, we settled on taping a pen to the end of the throwing arm, which happened to be the correct length we needed for the hook, but it also fastened the sling more securely, which resulted in the loop we tied around the hook to remain in place more consistently.

With these changes, our trebuchet design was able to successfully launch the rubber ball a distance of over 6 meters. The final design of our trebuchet, with the increased length of the nylon and added pen as our hook, can be seen in Fig. 4.

Fig 4. Finalized trebuchet design that includes the increased length of nylon, fastened to one end of the throwing arm and loosely attached to the hook (pen).

VI. CALCULATIONS

We modified a provided MATLAB script to calculate optimal lengths from the pivot point for each side of the main arm. The script ran 625 variations of lengths total, testing 25 different values for each parameter. The resulting graph is shown in Fig. 5. From this graph, we concluded that the optimal length of the short arm would be 0.3 m, and the optimal length of the long arm would be 0.4 m. These numbers did allow our final trebuchet design to achieve the 6-meter length requirement; however, the graph projects the launch distance at these numbers to be 0.2 meters. We were unable to ascertain why this was the case upon seeking help, but were advised that the figures would likely work despite the inaccurate launch distances. To check our results before building the trebuchet, we ran the script results through an online trebuchet calculator, which confirmed our design would satisfy the project requirements.

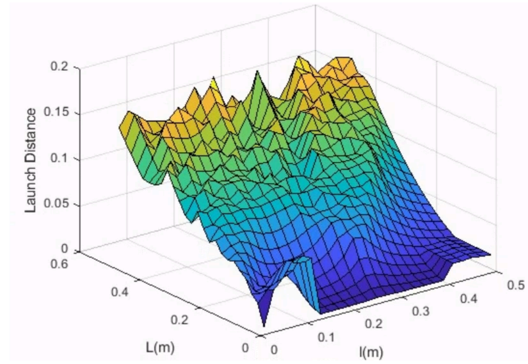

Fig 5. Graph of Long/Short Arm Length Versus Launch Distance.

VII. RESULTS AND DISCUSSION

For our final launch, we were able to launch the ball over the required 6-meter distance. As noted above, it took us several iterations to come to our finalized design, although we did use the same fundamental component lengths throughout our process. We may have been able to increase the distance more had we been able to resolve the issues with the MATLAB script outputting incorrect launch distance numbers. Future improvements on this design could involve a method of attaching the sling that is less prone to getting tangled, which was an issue we had with the nylon we chose. Additionally, alternative servo placements that allow the string to be released cleanly every time could produce more consistent results.

VIII. CONCLUSION

Designing, programming, and building a trebuchet helped us understand the difference between designing on paper and building a functioning machine. While the counterweight trebuchet is a relatively simple design, a consistent projectile launch of 6 meters, especially with a servo, was more challenging than we initially predicted.

Through much trial and error and some human ingenuity, we were able to launch consistently and successfully. Additionally, we gained comfort with MATLAB and Arduino, giving us experience with the programming and mechanical sides of this project.

REFERENCES

Chevedden, P. E. (2000). The Invention of the Counterweight Trebuchet: A Study in Cultural Diffusion. *Dumbarton Oaks Papers*, *54*, 71–116. https://doi.org/10.2307/1291833

Instructables. (2023, April 30). *Servo Driven Counterweight Trebuchet*. Instructables.

https://www.instructables.com/Servo-Driven-Counterweight-Trebuchet/

Olson, E., & Olson, P. (n.d.). *Virtual Trebuchet: A Web Based Trebuchet Simulator*. Virtualtrebuchet.com. https://virtualtrebuchet.com/

Ripcord. (2025). *Tennis Ball Trebuchet Plans*. Mrugala.net. http://medieval.mrugala.net/Armes%20de%20siege/Tennis%20Ball%20Trebuchet%20Plans/Tennis%20Ball%20Trebuchet%20Plans.htm

Spenko, M. (2019). *Trebuchet Part 1*. Mechanical Design. https://mechanicaldesign.iit.edu/trebuchet-part-1.html