Computationally and Physically Optimizing the Range of a Trebuchet

Charlie Bushman, Kyle Fraser-Mines, Jack Heinzel, and Axel Stahl
Carleton College, Department of Physics, Northfield, MN 55057
(Dated: April 10, 2021)

We explore the optimization of range for a trebuchet made of K'nex using physical and computational models. Lagrangian mechanics and Mathematica allows us to numerically calculate the range for various combinations of the four parameters: fulcrum location, length of the sling, presence of wheels, and initial hinged counterweight angle. Our computational model predicted the greatest range when the fulcrum is 0.09 m from the center of the throwing arm towards the counterweight, the sling is 0.49 m long, when the counterweight is approximately vertical, and when the body of the trebuchet is on wheels. Adjusting our trebuchet to match these optimal values increased the range from 5.2 m to 8.0 m, but could have theoretically increased it more.

I. INTRODUCTION

The trebuchet was a powerful medieval siege weapon, and was capable of throwing heavy stones as far as 300 meters.[2] We constructed a much smaller model capable of throwing a few grams about 8 meters. Despite being somewhat less menacing than the full scale models, there is a lot of analysis that can be done to maximize the efficiency of our smaller model, from which one can extract techniques to optimize a full scale trebuchet. We model our trebuchet using Lagrangian mechanics and a computational tool.

All of our analysis uses range as the quantity to maximize, and in most cases, for computational simplicity, we assume optimizing variables independently yields the *global* range maximum. In cases where the relationship between the variables could be strong, we optimize by finding the most effective pair of values for the two variables. The only case where it seems that optimizing the variables independently of each other might greatly miss the true optimal pair of values is the fulcrum position and sling length. We try to avoid any more of this, such as explicitly searching for the global maximum by varying four variables (for example, the four variables we have chosen to examine) together means order four computation time.

A. Lagrangian Techniques

We have four generalized coordinates that describe the trebuchet and map to Cartesian space ϕ, θ, σ, and r, as seen in Fig. 1. We map these four values to Cartesian space by

$$x_1 = r[t] + l_2 \cos\phi[t] - (\tfrac{l_1}{2} + d)\cos\theta[t] \tag{1a}$$
$$x_2 = r[t] - d\cos\theta[t] \tag{1b}$$
$$x_3 = r[t] + (\tfrac{l_1}{2} - d)\cos\theta[t] + l_3\cos\sigma[t] \tag{1c}$$
$$h_1 = h - (\tfrac{l_1}{2} + d)\sin\theta[t] - l_2\sin\phi[t] \tag{1d}$$
$$h_2 = h - d\sin\theta[t] \tag{1e}$$
$$h_3 = h + (\tfrac{l_1}{2} - d)\sin\theta[t] - l_3\sin\sigma[t], \tag{1f}$$

where (x_1, h_1), (x_2, h_2), and (x_3, h_3) correspond to the Cartesian coordinates of m_1 (the projectile), m_2 (the center of the rod), and m_3 (the counterweight) respectively. Note that $r[t]$, $\theta[t]$, $\phi[t]$ and $\sigma[t]$ are the four generalized coordinates, and will vary with time.

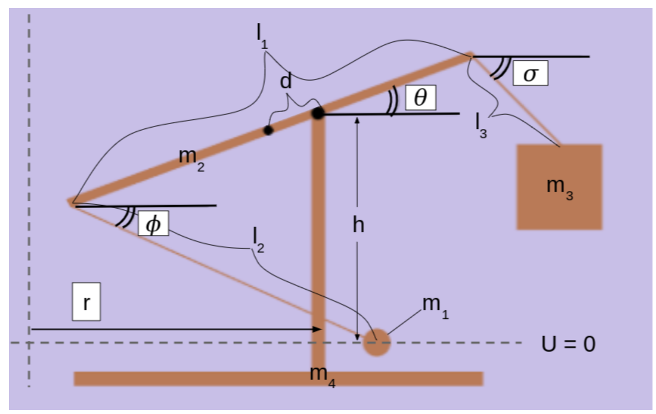

FIG. 1. A schematic drawing of our trebuchet model. r, θ, ϕ, and σ are functions of time and are the generalized coordinates highlighted in white. The other values are constants specific to the trebuchet model.

The Lagrangian is calculated from the kinetic energy T and potential energy U and, until the launch mass is lifted off the slide, the Lagrangian multiplier λ and constraint function G as well, as

$$L = T - U + \lambda G \tag{2}$$

and after the launch mass is lifted off the slide as

$$L = T - U, \tag{3}$$

where

$$\begin{aligned} T =& \tfrac{1}{2}m_1(\dot{x}_1^2 + \dot{h}_1^2) + \tfrac{1}{2}m_2(\dot{x}_2^2 + \dot{h}_2^2) + \\ & \tfrac{1}{2}m_3(\dot{x}_3^2 + \dot{h}_3^2) + \frac{1}{12}m_2 l_1^2\dot{\theta}^2, \\ U =& m_1 g h_1 + m_2 g h_2 + m_3 g h_3, \\ G =& h - (\tfrac{l_1}{2} + d)\sin\theta[t] - l_2\sin\phi[t] = 0. \end{aligned} \tag{4}$$

Subbing Eqs. 1 into Eq. 4 and into Eqs. 3 and 2, we get a pair of Lagrangians in terms of the generalized coordinates.

B. Computational Techniques

We use Mathematica v11.3 as the computational tool to evaluate this Lagrangian and the equations of motion, which we use with the Euler Lagrange equation to give us the equations of motion in terms of the initial conditions for each of our generalized coordinates

However, we have two Lagrangians and therefore two sets of equations of motion: one set for when the launch mass rests on the slide, with supportive normal force, and one for when the launch mass is lifted off, with no normal force. We transition between these two equations of motion by first finding the time when the normal force (represented in the Lagrangian (Eq. 2) by λ) becomes zero, which we can do using Mathematica's computational techniques, and using the instantaneous states of each of the generalized coordinates at this time as the initial conditions for the second set of equations of motion.

Mathematica can numerically approximate solution points to the equations of motion, then calculate an interpolating function between each point, a piecewise function of third degree polynomials.[4] This very accurately approximates solutions to the equations of motion.

To calculated the range, we first make a few assumptions. First, we assume that the time when the sling slips off the throwing arm - and therefore the time when the projectile is launched - is when the angles $\theta[t]$ and $\phi[t]$ are equal. Second, we assume the projectile is launched from zero height, and third, we assume that the only significant force on the projectile after launch is gravity from the earth.

We determined that these were reasonable assumptions for three separate reasons. For the first, we use slow motion video capture, and we see that the sling slipped off the throwing arm just about when the angles $\theta[t]$ and $\phi[t]$ are equal for a variety of different parameter values and initial conditions (see Appendix B). Second, though the projectile is not launched from zero height, it becomes very complex to include initial height into the range calculations, and the height of the trebuchet is negligible compared to the ranges we expect to throw. The third assumption is certainly reasonable because other forces, such as air resistance, buoyancy, or gravity from other bodies, are extremely small compared to the gravitational force from the earth.

Under these assumptions the range x is given by

$$x = \frac{v_0^2}{g}\sin[2\phi_0], \tag{5}$$

where v_0 is the launch speed, ϕ_0 is the initial launch angle above the horizontal, and g is gravity. We then must numerically calculate the velocity v_0 at the time of release by finding the norm of the rate of change of the x and y coordinates from Eq. 1a and Eq. 1d.

II. OPTIMIZATION OF FULCRUM POSITION

Our model kit had limited resources, so we cannot arbitrarily vary the magnitudes of all the parameters. Indeed, structural stability and material rigidity, as well as economic cost, ultimately also limit the scale of a full size trebuchet. Optimization of internal parameter values is something every trebuchet engineer should therefore consider. We can change internal parameter values, such as the location of the fulcrum. It is especially clear that the location of the fulcrum can have a dramatic effect on the range of the projectile. The placement of the fulcrum on the arm is important to get the maximum range possible. We use Mathematica to model the range for various fulcrum positions d.

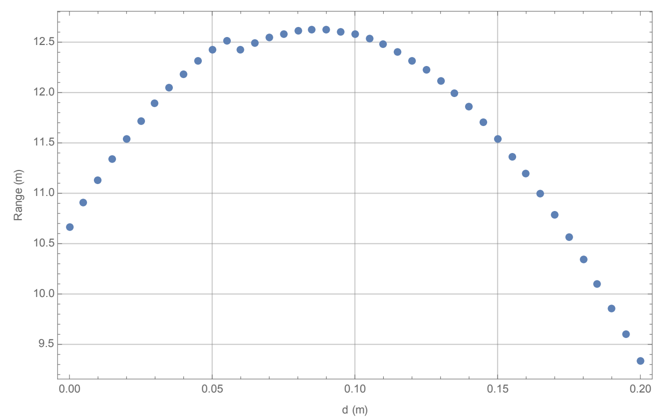

FIG. 2. A plot of calculated range vs fulcrum position d. Note that calculated range is maximized around $d \approx 0.09$m.

The range curve as a function of d has one local maximum between $d = 0$m and $d = 0.20$m at $d \approx 0.09$m. The peak is quite flat and wide near $d = 0.09$m, especially when considering that the range for $d = 0.20$m is still about 9m, not much smaller than the range for $d = 0.09$m. This is quite good for us, considering we operate using finite precision. If there were an extremely high maximum for some special value of d, however the peak was very narrow and difficult to obtain, then it would not be a good choice for d in our trebuchet. That is, we are glad that optimizing d does not require extreme precision.

There is also a strange discontinuity between $d = 0.055$m and $d = 0.06$m. (It is small and does not change where the maximum is, so it doesn't effect the final results.) When we focus in on this discontinuity we see that the behavior is still not smooth on a much smaller scale, indicating Mathematica may be having some difficulty. This may be from the FindRoot[4] function iterating a different number of times (the default is 15 iterations, though sometimes it may stop sooner due to a variety of reasons[4]), which could explain why the discontinuity is relatively small. If the discontinuity was resulting from FindRoot finding a different solution to where $\phi[t] = \theta[t]$, then the discontinuity would likely be much larger, and the behavior before and after the discontinuity be dramatically different. Indeed, given that the relative behavior before and after the discontinuity is essentially identical is very reassuring that this discontinuity can be ignored.

III. RELEASE VELOCITY AS A FUNCTION OF SLING LENGTH

Using a sling on the trebuchet allows us to throw the projectile much further. The sling makes the throwing arm virtually longer without actually increasing the mass or the length of the beam[2]. This adds a significant length to the arm of the trebuchet and increases the range of the projectile. Without the sling, the release velocity depends only on the length of the throwing arm and its angular speed. With a sling, the release velocity (and therefore the range) of the projectile depends on the length of the throwing arm, the length of the sling, and their angular velocities. In short, a sling allows a more efficient energy transfer to the projectile. But how long should the sling be to maximize this efficiency?

To determine the optimal sling length for the trebuchet we use an adjusted Mathematica function to optimize our fulcrum position. The program runs through our trebuchet simulations for each sling length over a reasonable region and determines the range of the projectile based on projectile's release angle and velocity. These varying sling lengths are plotted against the range of the projectile as seen in Fig. 3.

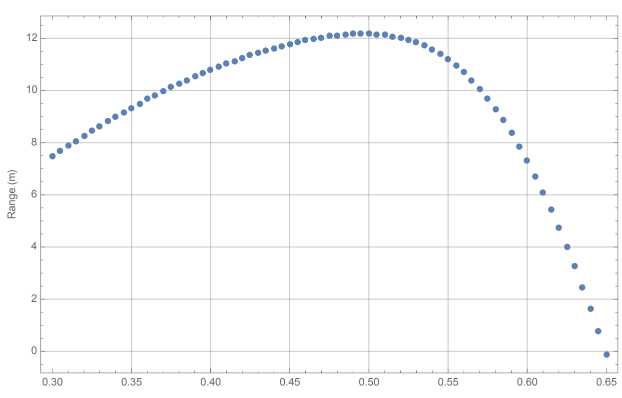

FIG. 3. A plot of the calculated range of the projectile vs the sling length. The longest range for the projectile occurs when the length of the sling is 0.49 m. The peak is relatively wide meaning small deviations from the optimal sling length will only change the range of the projectile minimally.

In Fig. 3, the range of the projectile is maximized when the sling is between 0.45 and 0.53 meters long. In this range, the release angle of the projectile was approximately $\pi/4$ above the horizontal which corresponds to the maximum range once the projectile is released from the trebuchet. At $l_2 = 0.49$m, the range of the projectile is at an absolute maximum, holding all other parameter values fixed. Additionally this maximum is relatively wide, which is helpful when adjusting out trebuchet to find the optimal value as the sling length could be slightly off of the optimal length of 0.49 m and still reach relatively the same range when the projectile is thrown.

A. Combined Effects of Fulcrum Position and Sling Length

While we calculated the optimal sling length and fulcrum position separately, research has shown that optimal sling length is equal to the distance from the fulcrum to the end of the launching arm[1] (note this author, DB Siano, defines "optimal" differently than we do; he defines it as efficiency of energy transfer). This indicated that different fulcrums positions may have different optimal sling lengths associated with them. So, the range of the trebuchet as a function of the fulcrum position and the range of the trebuchet as a function of the sling length may be related. In such a system, there is no guarantee that optimizing the variables independently will yield the global maximum of the pair of the variables.

If we are to find the global maximum, we sample a wider two dimensional array of values of the fulcrum position and the sling length. We run a nested loop to vary these two values together, creating a three dimensional plot of ranges as a function of the fulcrum position and the sling length.

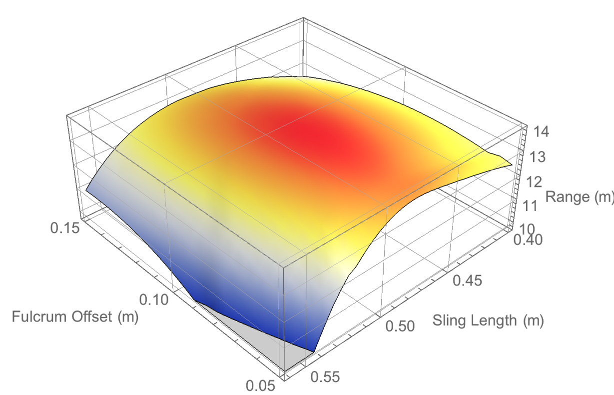

FIG. 4. The longest range occurs at $(d, l_2) \approx (0.09\text{m}, 0.49\text{m})$. The width of the maximum in both dimensions is of note, meaning small deviations from the optimal pair does not result in a large change in range.

Surprisingly, the global maximum is indeed the maximum we found optimizing the two variables independently. Furthermore, the three dimensional plot disagrees with Siano's claim, as the optimal value for sling length for any value of the fulcrum position between $d = 0.05$m and $d = 0.15$m is about $l_2 \approx 0.49m$. However, for the *optimal* pair of fulcrum position and sling length, the distance between the fulcrum and the end of the throwing arm is 0.48m, and this approximately matches the optimal sling length, in agreement with Siano.

In addition, the large width of the maximum in both dimensions is of note, meaning small deviations from the optimal pair of fulcrum position and sling length does not result in a drastically different range. As seen in Fig 4 above, any fulcrum position between $d = 0.07$ m and $d = 0.12$ m combined with any sling length between 0.45 m and 0.51 m will give relatively the same range for the projectile. Given that a trebuchet is a chaotic system[2], this is a nice surprise.

IV. THE EFFECT OF WHEELS ON RELEASE VELOCITY

Research has shown[2] that trebuchets with hanging counterweights and trebuchets with wheels can each throw projectiles further than trebuchets without these respective features, but he did not cover trebuchets with hanging counterweights and wheels.

We examine if a trebuchet with wheels and a hanging counterweight is able to launch projectiles further than trebuchets that lack wheels. We decide to compare the release velocity of the projectiles instead of the range of the trebuchets because we also want to physically test this question, and the tracker application we used can easily measure the velocity of an object, as well as allowing us to confirm our assumption that the sling releases the projectile when the angle of the throwing are and sling are equal (see Appendix B).

We first answered our question computationally, using our Mathematica notebook. We already have a model of our wheeled trebuchet, so we alos needed to model a stationary trebuchet independently. Formally, this is done rather easily: by removing the general coordinate r[t] from Eqs. 1 and therefore the Lagrangian. Comparing these two models, we find the wheeled trebuchet should throw a projectile farther. We calculate our wheeled trebuchet to launch a projectile with a release velocity of 12.60m/s, while our simulated immobile trebuchet only launched a projectile with a release velocity of 11.62m/s

Trebuchet Type	Computed Velocity	Real Velocity
Wheels	12.60 ms^{-1}	$9.59 \pm 0.11\text{ms}^{-1}$
No Wheels	11.62 ms^{-1}	8.79 ± 1.19 ms^{-1}

TABLE I. The computationally calculated and experimentally determined release velocities of projectiles launched from wheeled and immobile trebuchets.

After calculating the theoretical release velocities of the projectiles, we use the tracker application to collect data from slow-motion videos of our trebuchet, when it was both wheeled and immobile. We ran three trials for each version of our trebuchet, and determined error empirically from the differences between successive frames.

We originally expected our trebuchet's sling and projectile to have a noticeably higher velocity, when the trebuchet was wheeled because the movement of the trebuchet should add some momentum to the sling and projectile in the x-direction. Our theory seemed to be confirmed by our computational model, as it showed the wheeled trebuchet with a clearly faster release velocity.

Upon experimentally testing our trebuchet we find the differences in average release velocity between our trebuchet while wheeled versus while immobile to be less significant. Over three tests we found the average release velocity of a wheeled trebuchet to be 9.59 ± 0.11m/s while the average release velocity of our immobile trebuchet was 8.79 ± 1.19m/s. The uncertainty of the average release velocity of an immobile trebuchet is large enough that it overlaps with the average velocity of the wheeled trebuchet. This means we cannot confidently say that using wheels improve our trebuchet.

Despite our computational model suggesting that wheels improves the performance of our trebuchet, our final trebuchet has no wheels. This is because the trebuchet became so heavy that any wheels we tried to attach would simply fall off. We did not attempt any drastic measures in attaching wheels to the machine because our experimental data also suggested that the actual advantage of wheels was minimal at best.

V. RELEASE VELOCITY AS A FUNCTION OF INITIAL COUNTERWEIGHT ANGLE

The hanging counterweight is a very important feature of the trebuchet and contributes to an increase in the range of the projectile as compared to a fixed counterweight. The initial angle of the counterweight is often overlooked but will help increase the range of the projectile if optimized. While varying a full scale trebuchet's initial counterweight angle is impractical, this turns out to have a significant effect on the range of the trebuchet.

In Fig. 5, there are two clear maxima: one at $\sigma \approx 1.7$ rad and one at $\sigma \approx 4.6$ rad. 1.7 rad corresponds to when the counterweight is hanging down and pulled a slightly in towards the throwing arm and 4.6 radians corresponds to when the counterweight is tilted slightly toward the throwing arm while standing vertically above the arm. Simulation shows that an initial counterweight angle of $\sigma = 4.6$ rad would require the counterweight to fall through the launch arm which is not physically possible. In addition, we are less inclined to trust Mathematica's numerical analysis in regions outside of $-0.5 < \sigma < 2.3$ because this behavior tends to be erratic and sometimes nonphysical (ie. negative ranges or falling through the arm), indicating that Mathematica's numerical solving techniques may be finding different solutions.

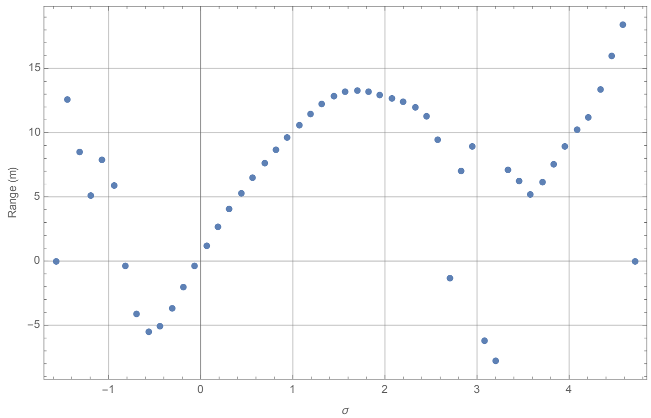

FIG. 5. There are two clear maxima: one at $\sigma \approx 1.7$ rad and one at $\sigma \approx 4.6$ rad. Simulation shows that an initial counterweight angle of $\sigma = 4.6$ rad would require the counterweight to fall through the launch arm.

The optimal initial counterweight angle being close to $\frac{\pi}{2}$ rad (hanging straight down) is encouraging for optimizing our model trebuchet. Also, since the peak on the graph is wide, we can simply let it hang at $\frac{\pi}{2}$ rad without losing a significant amount of the range we would get at the optimal counterweight angle.

VI. FINAL RESULTS AND DISCUSSION

Our computational model calculated optimal values for maximizing the range of a 10 gram projectile thrown by a 300 gram counterweight from an initial height of zero. For these values, $d = 0.09$m, $l_2 = 0.49$m, $\sigma_0 = 1.7$rad on a wheeled trebuchet are optimal for maximizing range (see Appendix C for all other parameter values).

In hindsight, we can provide further reasoning behind the applicability of our assumption for simplicity that the projectile is thrown from a height of zero. First, as we mentioned earlier, the initial height of the trebuchet is small compared to the ranges thrown, so it will ultimately be negligible. Second, the additional range accounting for the height of the launch is approximately the same for the optimal values, because they're all presumably launching at about 45^o, by virtue of being the optimal ranges. Since this extra range is a function of launch angle and height launched at, and height is constant, the excess range will be the same for each. While assuming the projectile launches from zero initial height may predict slightly shorter ranges than the trebuchet should might in reality, is a reasonable assumption to make to find the optimal parameters for maximizing range.

This argument rests on the peaks for our curves being relatively flat. For a parameter-range curve that has a thin peak, we could not assume that launches within a region around the peak have similar launch angles. We are pleased to see, therefore, that each of our parameter curves have relatively flat peaks.

A. Applying to our Model Trebuchet

When we adjusted our physical trebuchet to these parameter values, we saw a promising and significant increase in range. However, after a few test runs, the apparatus that made our counterweight around 300 grams broke and we lost about 100 g of mass, so our official range measurement only came out to be about 8 meters, far from the 12-13 meter ranges our computational model predicted. The larger trebuchet was also did not have wheels, as the body became too heavy to be supported on wheels, which, though we saw wasn't extremely important for range, could change where the optimum values for our parameters were, as could the change in counterweight mass. Not only did our trebuchet lose wheels and counterweight mass (both of which contribute positively to range), but also may have no longer been optimized.

Additionally, our model did not account for air resistance and bending in the trebuchet. These losses of energy may also have decreased the range, and may help explain why our trebuchet failed to throw as far as the computational model predicted.

ACKNOWLEDGMENTS

Specific thanks goes to Jay Tasson for facilitating our progress and answering our questions, to Bruce Duffy for introducing us to the Tracker program, and to our classmates for being attentive to our presentation.

[1] Donald B. Siano, "Trebuchet Mechanics", dimona@home.com, , (2001)

[2] Mark Denny, "Siege Engine Dynamics", Eur. J. Phys. **26** (561), , (2005)

[3] Zenos Christo "Analysis of the Optimum Fulcrum Position of a Trebuchet", Phys. Educ. **52** (013010), , (2017)

[4] Wolfram Research, Inc., Mathematica, Version 11.3, Champaign, IL (2018).

APPENDIX A

Charlie Bushman served as the code keeper for our group. Charlie helped build our initial trebuchet at the beginning of the project. After we had our Lagrangian in Mathematica Charlie worked with the code to create a program the we could enter our parameters into and adjust to optimize the variables we were testing. Charlie also helped look over the codes that the other group members created in testing each of their questions. Charlie also created code that looped through the fulcrum position and sling length to find the optimum pair and edited the existing code to make sure it worked and was accurate. Charlie wrote Section V of the paper on how the initial counterweight affects the range of the projectile. Charlie incorporated the results for his question into the presentation.

Kyle Fraser-Mines served as the trebuchet master for our group. In building our initial model for the trebuchet Kyle helped put the model together and helped made sure the trebuchet would work when it was launching the projectiles during our tests. Kyle created a section of code that would determine the sling length that leads to the furthest range of the projectile. Kyle helped Axel get experimental data when the trebuchet launched the projectile. Kyle rebuilt our model trebuchet using the optimal parameters found in our research in order to increase the range of our trebuchet. Kyle wrote Section III on the effect of sling length on the range of the projectile and helped write the subsection on how fulcrum position and sling length are related. Kyle incorporated the results for her question into the presentation.

Jack Heinzel served as the paper editor for our group. Jack helped build our model trebuchet at the beginning. Once we found the Lagrangian for our trebuchet system Jack put the system into Mathematica and created an animation showing us that our Lagrangian was accurate. Jack created a section of code that would determine the fulcrum position that leads to the furthest range of the projectile. Jack also worked with Charlie to create code that looped through the fulcrum position and sling length to find the optimum pair and edit the existing code to make sure it worked and was accurate. Jack wrote Section II on the effect of fulcrum position on the range of the projectile and helped write the subsection on how fulcrum position and sling length are related. While everyone helped write the introduction and the final results for the experiment Jack took the lead on this and made sure everything flowed smoothly. Jack incorporated the results for his question into the presentation.

Axel Stahl served as the talk coordinator for our group. Axel helped build our initial trebuchet at the beginning. Axel created a section of code that would determine how the presence of wheels would effect the range of the projectile. Axel also used the Tracker system to determine the experimental release velocity of the projectile for our trebuchet with and without wheels. Axel helped rebuild the model trebuchet to increase range. Axel wrote Section IV of the paper on how the presence of wheels affect the range of the projectile. Axel coordinated the presentation and created a template that the group could add results into for each of their respective questions.

APPENDIX B

FIG. 6. Tracker image of the release angle of the projectile. This is the frame right before the projectile is released. Note the projectile (and the sling that runs between the projectile and the end of the throwing arm) are almost perfectly aligned.

APPENDIX C

Mathematica notebook available on GitHub (https://github.com/Ulthran/Fun-With-Mathematica/blob/master/Trebuchet.nb).