Ricky Miranda

Professor Sato

Computational Physics (335)

May 3, 2020

Calculating the Advantageous Body Proportions for Powerlifting

Necessary Background Information and The Physical System:

We'll begin this paper by first explaining some necessary background information regarding the sport of Powerlifting, Anthropometrics, and the basic physics behind this project and what the physical system is comprised of.

The sport of Powerlifting is considered a strength sport (such as Olympic Weightlifting or Strongman, and not to be confused with sports such as Bodybuilding) that revolves around performing three maximal lifts: Squat, Bench Press, and Deadlift, on a barbell (depending on the lift, different types barbells can be used) in order to obtain a total which will determine if you are the winner of your weight class or not.

Anthropometry and anthropometrics is the study of the measurements and proportions of the human body. Something related to body proportions you may have heard of is that the ratio of our wingspan, when compared to our height, is almost 1:1 on average, i.e extending your arms out such that roughly a $90°$ angle is formed between your arm and torso at the arm pit is made, the measurement from your left middle fingertip to your right middle fingertip will roughly equate to your height. Another example which we will also use in this simulation is how the length of our head compares to our height. Our height is on average, 7.5 times the length of our head. The reason why this is important is because our parameters will depend on these body ratios to get proper body measurements to test in our simulation and will reflect possible human body proportions. These body ratios will pose as necessary restrictions for our simulation. For example, it is almost certain that there hasn't existed a human being who was 5 feet tall with a 6 foot wingspan. Including average body ratios across the human population will exclude these possible combinations from occurring.

Unfortunately, after a few hours of generally researching anthropometrics and possible body ratios, I wasn't able to find standard deviations of body ratios across a large enough sample size of humans so I had to come up with some values of my own based on my own body measurements as well as the lower and upper bounds. As a result, this could lead to errors in the possible body proportions that exist and the results of the simulation since the body ratios were used. The provided values or ratios are compared to the height of an individual and in the code, I specified lower and upper bounds to simulate across different proportions. The lower and upper bounds are $\pm\ 0.05h$, 5% of the height of an individual. We will calculate arm length a, femur length f, torso length t, and lower leg (the distance from the bottom of their foot to their knee) length l as follows:

$$\begin{aligned}
a &= \frac{h - bw}{2}, \\
f &= \frac{h}{4}, \\
t &= \frac{h}{4}, \\
l &= \frac{h}{3.25},
\end{aligned}$$

where h is the height, and bw is the biacromial width of an individual and is defined as $bw = \frac{h}{5}$. The constants are based on body ratios that I found that apply to myself. I will restate this once more for caution that, because I used formulas that I came up with based on my own body measurements and whatever information I could find, the results may not properly reflect the human population in general.

What we will be looking at in this paper is whether or not different body proportions are more advantageous than others, specifically for the deadlift. Simply put, performing a deadlift is when you lift a barbell off the ground to the level of your hips (or however far your arms reach, we will see the results of this later) and your torso is perpendicular to the floor. This is known as "locking out".

The assumptions that we will make for this simulation require that we do not take into consideration the vast and unique ways that one powerlifter can deadlift (such as “pulling” Sumo or Conventional, two different stances for deadlifting) and instead assume everyone will pull in a conventional stance exactly the same way, where the only differences between powerlifters are their body proportions. The path of the bar will also be assumed to undergo a straight line (known as straight bar path) in order to limit our results where work is done only in the vertical direction (if the bar path is not straight, then there is expended energy we must take into account, thus we assume that it is completely straight). Interestingly enough, a straight bar path is the most efficient way to deadlift because lifting a barbell in any other path that is not straight will result in having the bar travel a greater distance, no matter how small. This is known as energy leakage. While we may think the small distance is negligible, the heavier a barbell is loaded, the more energy that will be required to lift over that small distance.

We will visualize this system to be in a Cartesian plane with an x-axis in the horizontal direction and y-axis in the vertical direction. The physical system will consist of a single one-dimensional bar (expanding over the x-axis) undergoing two forces, the gravitational pull of the earth as a constant (which is a vector that points in the -y direction) and the force required to move the bar upward a certain distance (a vector pointing upward in the +y direction). The distance the bar must move is determined by a specific instance which we will call a “Powerlifter” which we will represent as a class that signifies an object. We will see this class in more detail later. One specific instance of a Powerlifter would simulate a unique lifter that is lifting the barbell at rest by calculating the work done based on their proportions and compare that calculation with other powerlifter objects of the same height and see which one results in having the least amount of work done. This would mean that the advantageous lifter is moving the prescribed weight a lesser distance and so can theoretically lift more weight to match the work done by the other powerlifters.

The work done will be calculated using the work formula, which is,

$$W = F \cdot d,$$

where $F = 9.81 \cdot m$, 9.81 is the gravitational constant of the earth, m is the mass of the object being lifted, and d is the distance the object has to travel.

The different parameters we will use for this simulation would be based on the body measurements, (as discussed earlier) of a unique lifter, which we will define as the lifter's height, and the lengths of their torso, femur, arm, and lower leg in meters. These parameters would then form a unique physical system or Powerlifter object that would simulate a unique lifter.

The Code:

For this section, I will include a compressed version of my code with only the internal methods (if any) and their input parameters. I will explain what part of my code does and how it plays a role into finding the answer we are looking for. As a reminder, we are looking for the advantageous body proportions for a certain height, across many heights. We first start with the Powerlifter class, which is defined as follows:

```
class Powerlifter{
    Powerlifter(height, arm, femur, torso, lowerLeg, mass):
    shoulderHeight():
    distanceToFloor():
    getProportions():
    getCalculatedHeight():
    workDone():
}
```

The Powerlifter class contains six methods, including the constructor. As said earlier, the input parameters are the height, arm, femur, torso, and lower leg lengths of the individual. We include another input parameter, mass, which is the mass of the object that is being lifted. The mass will be constant across all lifters for this simulation. We could've easily created a static variable for the mass but in case one wishes to see how much force is required to lift a certain weight, we leave it as an input parameter. The method, shoulderHeight(), calculates the height up to the shoulders (excluding the head and neck in one's total height) and returns that height. The method, distanceToFloor(), calculates the distance the bar has to travel from the floor to roughly the lowest point of the arms at lockout and returns that distance. Because the barbell is elevated due to the weights loaded on it, the barbell does not necessarily start at the floor but instead 0.21 meters off the floor using standard plates. That is taken into account in this method. The method, getProportions(), returns a string that provides the proportions of the unique instance of the powerlifter. The method, getCalculatedHeight(), is a method that returns the height of the powerlifter based on the sum of the provided body proportions and the height to head ratio of 1:7.5, as discussed earlier. This method will be used to verify that the proportions given to a specific powerlifter object adds up to the given height of a lifter. The method, workDone(), calculates the work done when lifting an object, with the specified mass given as a parameter, a certain distance, that distance being the value from distanceToFloor(). The next piece of code we have is the simulation itself, which is the bulk of the work and involves this next method.

```
simulateBestPowerlifters(minHeight, maxHeight):
    arrayHeights, arrayBest = simulatePowerlifters(minHeight, maxHeight)
    getBestPowerlifters(arrayHeights, arrayBest)
    total = totalNumberOfLifters(arrayHeights)
    totalHeights = maxHeight - minHeight
```

While it may not look like much, a lot of work is going into the method simulatePowerlifters(minHeight, maxHeight). What this method does is pose the restrictions described earlier in this paper and goes through a series of for loops to create many specific instances of Powerlifters and, depending on the range of heights inputted, can range in the tens of thousands of objects. The way it works is, you input a range of heights in centimeters and run the method, simulateBestPowerlifters(minHeight, maxHeight), and it will return all the “best” calculated proportions based on which powerlifter did the least amount of work for each height. You can see the code in more detail in the attached .py files.

As an example, I will show the output of this method for a range of 5 heights on the next page.

simulateBestPowerlifters(160,165)

returns

The body proportions of this powerlifter are: 1.6 m, 0.71 m, 0.43 m, 0.42 m, and 0.48 m for the lengths of height, arm, femur, torso, and lowerLeg, respectively.

The body proportions of this powerlifter are: 1.61 m, 0.71 m, 0.43 m, 0.42 m, and 0.48 m for the lengths of height, arm, femur, torso, and lowerLeg, respectively.

The body proportions of this powerlifter are: 1.62 m, 0.71 m, 0.42 m, 0.48 m, and 0.44 m for the lengths of height, arm, femur, torso, and lowerLeg, respectively.

The body proportions of this powerlifter are: 1.63 m, 0.72 m, 0.43 m, 0.44 m, and 0.48 m for the lengths of height, arm, femur, torso, and lowerLeg, respectively.

The body proportions of this powerlifter are: 1.64 m, 0.72 m, 0.42 m, 0.48 m, and 0.46 m for the lengths of height, arm, femur, torso, and lowerLeg, respectively.

The body proportions of this powerlifter are: 1.65 m, 0.73 m, 0.43 m, 0.46 m, and 0.48 m for the lengths of height, arm, femur, torso, and lowerLeg, respectively.

The total number of powerlifters simulated is 11959 across 5 different heights.

The Applications of this Simulation:

While this simulation may seem a bit trivial at first due the variety of ways that lifters pull, it wouldn't be difficult to create a more advanced and detailed program to account for such differences. Regarding the results, they were actually satisfying to see (long arms and almost 1:1:1 across torso, femur, and lower leg) because it is in fact true that such body proportions would be advantageous for deadlifting. Examples such as Lamar Gant and Cailer Woolam would fit into the categories of having the best proportions for deadlifting due to their long arm lengths in comparison with the rest of their body proportions. Another possible way of using this project's code is on a case-by-case basis. Individuals will be able to input their exact lengths of their body proportions as opposed to having a simulation guess what possible combinations of body proportions there exists within the human population. With the exact body proportions of an individual, we can see where they fall in the category of simulated powerlifters as a percentile if we were given enough information regarding the distribution of proportions across the powerlifting population.

You can even input your own body part lengths and create a Powerlifter() object and see how much work is done for a certain mass for someone with your exact proportions. As a final example, I will input my own height and body part lengths.

Powerlifter(1.85, .79, .49, .50, .54, 100)

Using the workDone() method, it states that the amount of work done when lifting 100 kg for a person with my body measurements is around 520 N·m.

Fortunately, because this project lined up with a passion of mine so well, I plan to work on this project further for the sake of my own and potentially other peoples' interests.