APPLICATIONS OF PHYSICS IN COMPUTING

Animation

Animation is a method of photographing successive drawings, models, or even puppets, to create an illusion of movement in a sequence. Because our eyes can only retain an image for approx. 10^{th} of a second, when multiple images appear in fast succession, the brain blends them into a single moving image. Animation is t he process of displaying still images in a rapid sequence to create the illusion of movement.

The Taxonomy of Physics-Based Animation Methods

At the highest level, the field of physics-based animation and simulation can roughly be subdivided into two large groups:

1. Kinematics is the study of motion without consideration of mass or forces.
2. Dynamics is the study of motion taking mass and forces into consideration.

kinematics and dynamics come in two flavors or subgroups:

1. Inverse is the study of motion knowing the starting and ending points.
2. Forward is the study of motion solely given the starting point.

Frames

A frame is a single image in a sequence of pictures. A frame contains the image to be displayed at a unique time in the animation. In general, one second of a video is comprised of 24 or 30 frames per second also known as FPS. The frame is a combination of the image and the time of the image when exposed to the view. An extract of frames in a row makes the animation.

Frames per Second

Animation shot on film and projected is played at 24 frames per second. Animation for televisionin Europe, Africa, the Middle East and Australia is played at 25 frames per second.

Sl. No.	System	Frames Per Second
1	PAL (Australia, Middle East, Africa)	25
2	NTSC(America, West Indies, Specific Rim Countries)	30

An animated film with 25 frames per second is played on television at 24 frames per second would result in a black bar rolling up the screen. Then Digital Converts are to be used to transfer one speed of film to another speed of video. The most important thing to find out when animating something is what speed the animation will be played back at.

Size and Scale

The size and scale of characters often play a central role in a story's plot. What would Superman be without his height and bulging biceps? Some characters, like the Incredible Hulk, are even named after their body types.

We often equate large characters with weight and strength, and smaller characters with agility and speed. There is a reason for this. In real life, larger people and animals do have a larger capacity for strength, while smaller critters can move and maneuver faster than their large counterparts. When designing characters, you can run into different situations having to do with size and scale, such as:

1. Human or animal-based characters that are much larger than we see in our everyday experience. Superheroes, Greek gods, monsters,
2. Human or animal-based characters that are much smaller than we are accustomed to, such as fairies and elves.
3. Characters that need to be noticeably larger, smaller, older, heavier, lighter, or more energetic than other characters.
4. Characters that are child versions of older characters. An example would be an animation featuring a mother cat and her kittens. If the kittens are created and animated with the same proportions and timing as the mother cat, they won't look like kittens; they'll just look like very small adult cats.

Proportion and Scale

Creating a larger or smaller character is not just a matter of scaling everything about the character uniformly. To understand this, let's look at a simple cube. When you scale a cube, its volume changes much more dramatically than its surface area. Let us say each edge of the cube is 1 unit length. The area of one side of the cube is 1 square unit, and the volume of the cube is 1 cubed unit. If you double the size of the cube along each dimension, its height increases by 2 times, the surface area increases by 4 times, and its volume increases by 8 times. While the area increases by squares as you scale the object, the volume changes by cubes.

Wight and strength

Body weight is proportional to volume. The abilities of your muscles and bones, however, increase by area because their abilities depend more on cross-sectional area than volume. To increase a muscle or bone's strength, you need to increase its cross- sectional area. To double a muscle's strength, for example, you would multiply its width by $\sqrt{2}$. To triple the strength, multiply the width by $\sqrt{3}$. Since strength increases by squares and weight increases by cubes, the proportion of a character's weight that it can lift does not scale proportionally to its size.

Let us look at an example of a somewhat average human man. At 6 feet tall, he weighs 180 pounds and can lift 90 pounds. In other words, he can lift half his body weight. If you scale up the body size by a factor of 2, the weight increases by a factor of 8. Such a character could then lift more weight. But since he weighs more than 8 times more than he did before, he can not lift his arms and legs as easily as a normal man. Such a giant gains strength, but loses agility.

Motion and Timing in animations

Introduction to Motion :

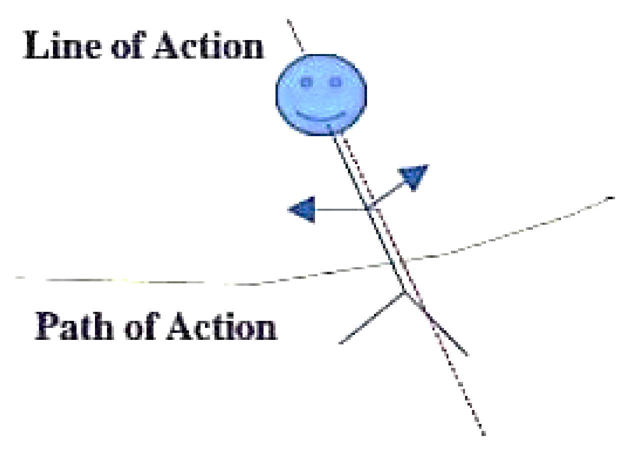
Motion is an essential component in games and animations. The motion is governed by the newtons laws and kinematic equations. When animating a scene, there are several types of motion to consider. These are the most common types of motion:

1. Linear
2. Parabolic
3. Circular
4. Wave

Motion and timing go hand in hand in animation.

Motion Lines and Paths

Individual drawings or poses have a line of action, which indicates the visual flow of action at that single image. Motion has a path of action, which indicates the path along which the object or character moves. The path of action refers to the object's motion in space. While it can help show timing, its primary function is to see the direction and path of the motion, and not necessarily its timing.



Timing

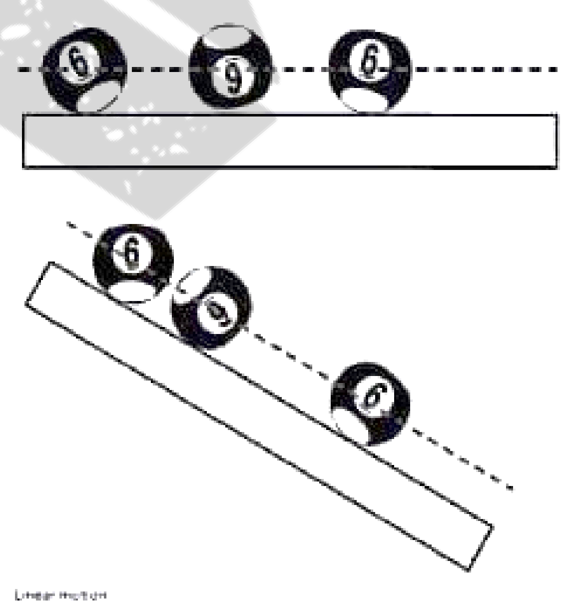
The timing is the choice of when something should be done; the regulation of occurrence and pace to achieve a desired effect. Animators have the ability to move forward and backward in time to place objects when and where they are to be.

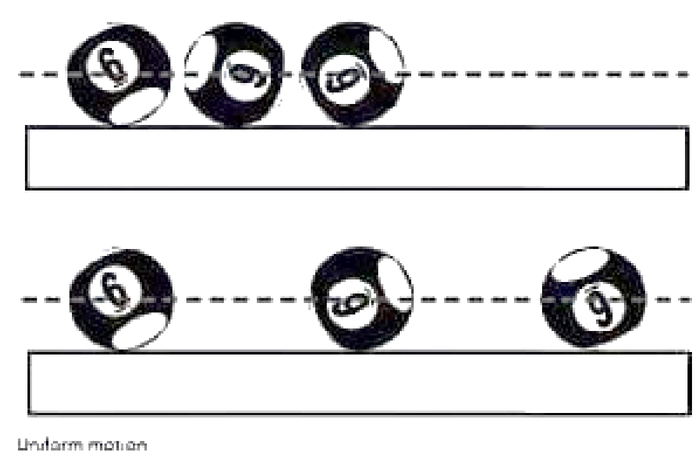
Timing Tools

In animation, timing of action consists of placing objects or characters in particular locations at specific frames to give the illusion of motion. Animators work with very small intervals of time; most motion sequences can be measured in seconds or fractions of seconds. Frame intervals between keys are usually smaller than one second.

Linear Motion Timing

Linear motion refers to motion in a straight line, always in the same direction. An object moving with linear motion might speed up or slow down as it follows a linear path. A heavy ball rolling on a table or incline is an example of linear motion. The all is rotating, but its center of gravity follows a linear path. A heavy ball rolling on a table or incline is an example of linear motion. The ball is rotating, but its center of gravity follows a linear path.





Uniform Motion Timing

When uniform motion occurs, the net force on the object is zero. Net force is the total of all forces added up. There might be several forces acting on the object, but when both the magnitude and direction of the forces are added up, they add up to zero. Uniform motion is the easiest to animate because the distance the object travels between frames is always the same. Uniform motion is a type of linear motion with constant speed and no acceleration or deceleration. The object moves the same distance between consecutive frames. The longer the distance between frames, the higher the speed.

Slow in and Slow out

When motion is accelerating or decelerating, we refer to this type of motion as a slow in or slow out. This type of motion is sometimes called ease in or ease out. In this book, we use the hyphenated forms slow-in and slow-out for easier understanding.

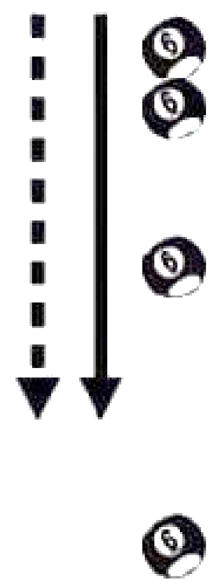
1. Slow in, ease in—The object is slowing down, often in preparation for stopping.
2. Slow out, ease out—The object is speeding up, often from a still position.

The term slow out can be confusing, since it essentially means "speed up." one can think of slow out as the same as ease out, as in easing out of a still position and speeding up to full speed.

For example, a ball rolling down an incline or dropping straight down is slowing out, as it goes from a still position or slow speed to a fast speed. A ball rolling up an incline is slowing in.

Slowing out: speeding up.

Slowing in: slowing down.



Acceleration Timing

Timing for acceleration can be calculated very accurately when the net force being exerted is constant. Let's take a look at the forces and how they can be used to calculate the animation's timing.

Constant Forces

A constant force is a force that doesn't vary over time. Examples of constant forces include:

1. Gravity pulling an object to the ground
2. Friction bringing an object to a stop

Constant force and Acceleration

Constant forces result in constant acceleration. Because the acceleration is constant, we can figure out the timing for such sequences using a few principles of physics.

The resulting acceleration depends on the direction of the force and motion, if there is any motion at all to begin with.

1. When constant net force is applied to an unmoving object, the result is acceleration.

2. When constant net force is applied to a moving object in the same direction as the motion, the result is acceleration.
3. When constant net force is applied in the direction opposite the existing motion, the result is deceleration (acceleration in the opposite direction).

Forces Exerted by Characters

Forces exerted by people's bodies are rarely constant throughout an entire motion. For the purposes of animation, however, one can break the character motion into short time segments and consider each of these segments to be responding to constant net force. This will make it easier for one to calculate the timing for each individual segment.

As an example, let's look at the push for a jump. The force a character exerts during the push is somewhat constant, and the timing is very short (less than half a second). In such a case the timing for a constant force is an excellent starting point, and in most cases will do the job as is.

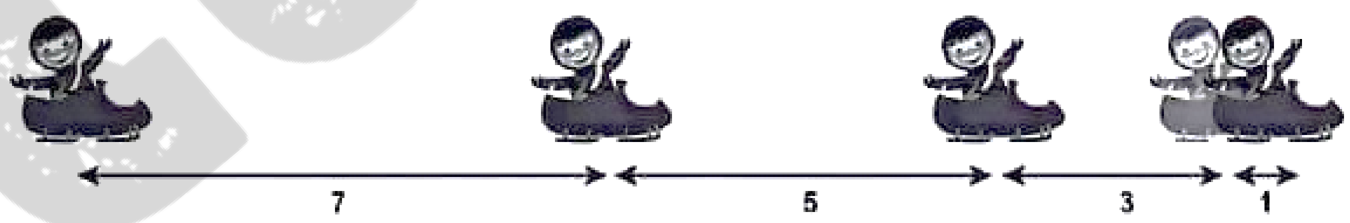
A character walking and pushing a rock is not exerting a constant force throughout the entire sequence, but during each short part of the walk cycle the net force could be considered to be a different constant value.

The Odd Rule

When acceleration is constant, one can use the Odd Rule to time the frames. With this method, one calculate the distance the object moves between frames using a simple pattern of odd numbers. Between consecutive frames, the distance the object moves is a multiple of an odd number. For acceleration, the distance between frames increases by multiples of 1, 3, 5, 7, etc.

Rocket speeding up using the Odd Rule.

For deceleration, the multiples start at a higher odd number and decrease, for example 7, 5, 3, 1.



Sled coming to a stop using the Odd Rule.

The Odd Rule is a multiplying system based on the smallest distance traveled between two frames in the sequence. For a slow-out, this is the distance between the first two frames; for a slow-in, it's the distance between the last two frames. This distance, the base distance, is used in all Odd Rule calculations.

Odd Rule Multipliers

The Odd Rule in its simplest form, as described above, is just one way to use it. For example, one can instead calculate the distance from the first frame to the current frame and use these distances to place the object on specific frames.

	Multiply by base distance to get distance between:	
Frame #	Consecutive frames	First frame and this frame
1	n/a	0
2	1	1
3	3	4
4	5	9
5	7	16
6	9	25
7	11	36

calculating the distance for a large number of frames and a chart like this isn't practical, one can figure out the odd number multiplier for consecutive frames with this formula:

Odd number multiplier for consecutive frames = ((frame # – 1) * 2) – 1

In the charts above, note that the distances in the last column are squared numbers: 4 = 22, 9 = 32, 16 = 42, and so on. One of the benefits of the Odd Rule is one can calculate the total distance traveled from the start point to the current frame with the following formula:

Multiplier for distance from first frame to current frame = (current frame # – 1) 2

When setting the keys, one can use either the consecutive key multipliers or total distance multipliers but need to Choose the one that's easiest to use for the animated sequence.

Odd Rule Scenarios

Here are a few different scenarios for calculating the distance an object travels between keys in a slow-in or slow-out.

Base Distance Known Speeding up

If the object is speeding up, the first frame distance is the base distance. If one knows the base distance, figuring out the distance the object travels at each frame is pretty straightforward. Just multiply the base distance by 3, 5, 7, etc. to get the distances between consecutive frames, or use squares to multiply the base distance to get the total distance traveled on each frame.

Base Distance Known Slowing Down

Suppose one wants an object to slow down, and one knows the distance between the last two frames before it stops. For slow-ins, the base distance is the distance between the last two frames. The solution is to work backward, as if the object were speeding up in the opposite direction. Working backward, multiply the base distance by 3, 5, 7, etc. to get the distances between each previous frame in the sequence.

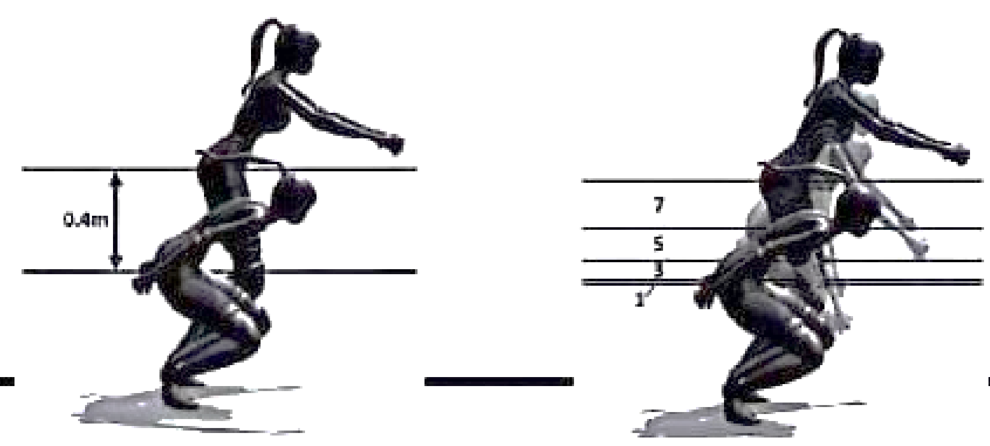
Total Distance and Number of Frames Known, Speeding Up

If one wants know the total distance and the total number of frames, one can find the base distance with this formula:

Base distance = Total distance/(Last frame number – 1) 2

Suppose there is a jump push (takeoff) with constant acceleration over 5 frames, and the total distance traveled is 0.4m. Using the formula above, we find the base distance.

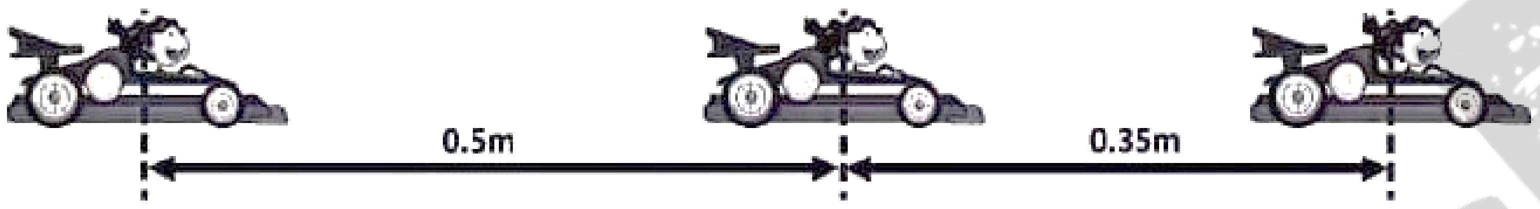
Base distance = 0.4m/(5 – 1) 2 = 0.4m/16 = 0.025m



Using the base distance, one can calculate the distances between each frame. If one adds up the distances traveled, one will find that they add up to exactly 0.4m.

First Key Distance Known Slowing Down

Suppose one has a moving object that one wants to slow down, and one has set the first frame of the slow-in to give an idea of the pacing for the sequence. In this case, one can consider that the distance the object moved between the last two frames before the slow-in is part of the calculation— the distance between them becomes the fi rst frame distance, and the first slow-in frame becomes the second frame in the sequence.



One feature of the Odd Rule is that the base distance is always half the difference between any two adjacent distances. To find the base distance, one can simply calculate:

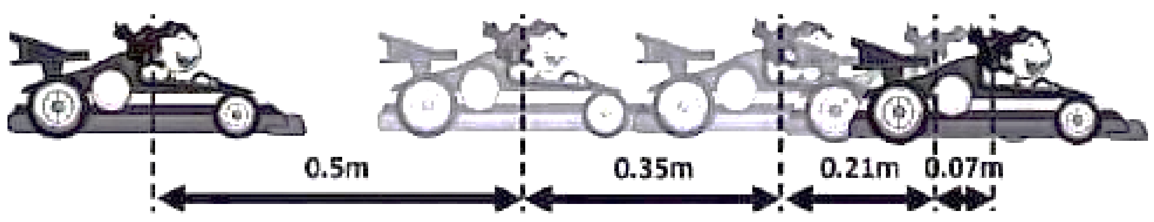
(0.5m – 0.35m)/2 = 0.07m

To figure out how many frames are in the slow-in, divide the first distance by the base distance to find out which odd number it corresponds to.

0.5/0.07 = 7

This means the first distance corresponds to 7 in the 7, 5, 3, 1 sequence, making the sequence four frames long. Now ibe can work back the other way, multiplying the base distance by odd numbers to get the distances for the rest of the slow-in frames.

Frame #	Consecutive frame multiplier	Distance from previous frame
1	7	7 * 0.07m = 0.5m
2	5	5 * 0.07m = 0.35m
3	3	3 * 0.07m = 0.21m
4	1	1 * 0.07m = 0.07m



Motion Graphs

A motion graph plots an object's position against time. If one is using animation software, understanding and using motion graphs is a key skill in animating anything beyond the simplest of motions. If one is drawing the animation, drawing motion graphs before animating can help one to visualize the motion. On a motion graph, the time goes from left to right across the bottom of the graph, while the object's position is plotted vertically against the time. Each axis in 3D space (X, Y, Z) has its own line showing the object's position along that axis. At the very least, one will need to understand the types of lines in a motion graph and what they represent in terms of visible motion. one can also look at motion graphs to get a better understanding of any difficulties one is having with the timing or action.

Examples of Character Animation

Jumping and Walking

Jumping

A jump is an action where the character's entire body is in the air, and both the character's feet leave the ground at roughly the same time. A jump action includes a takeoff, free movement through the air, and a landing.

Parts of Jump

A jump can be divided into several distinct parts:

- **Crouch—**A squatting pose taken as preparation for jumping.
- **Takeoff—**Character pushes up fast and straightens legs with feet still on the ground. The distance from the character's center of gravity (CG) in the crouch to the CG when the character's feet are just about to leave the ground is called the push height. The amount of time (or number of frames) needed for the push is called the push time.
- **In the air—**Both the character's feet are off the ground, and the character's center of gravity (CG) moves in a parabolic arc as any free-falling body would. First it reaches an apex, and then falls back to the ground at the same rate at which it rose. The height to which the character jumps, called the jump height, is measured from the CG at takeoff to the CG at the apex of the jump. The amount of time the character is in the air from takeoff to apex is called the jump time. If the takeoff pose and the landing pose are similar, then the jump height and jump time are about the same going up as they are going down.
- **Landing—**Character touches the ground and bends knees to return to a crouch. The distance from the character's CG when her feet hit to the ground to the point where the character stops crouching is called the stop height. The stop height is not always exactly the same as the push height.

Calculating Jump Actions

When working out the timing for a jump, one will need to first decide on:

1. Jump height or jump time
2. Push height
3. Stop height
4. Horizontal distance the character will travel during the jump

From these factors, one can calculate the timing for the jump sequence.

Calculating Jump Timing

When planning the jump animation, the most likely scenario is that you know the jump height, expressed in the units you are using for the animation (e.g., inch or cm).

Placement and timing for frames while the character is in the air follow the same rules as any object thrown into the air against gravity. Using the tables in the Gravity chapter (or an online calculator), one can figure out the jump time for each frame. Look up the amount of time it takes an object to fall that distance due to gravity, and express the jump time in frames based on the fps one is using.

Example:

Jump height = 1.2m

Jump time for 1.2m = 0. 5 seconds

Jump time at 30fps = 0. 5 * 30 = 15 frames

Jump Magnification

When calculating the remainder of the timing for the entire jump action, you can use a factor called jump magnification (JM). The JM can be used to calculate the push timing and stop timing.

The JM is the ratio of the jump height to the push height.

$$JM = \frac{Jump\ Height}{Push\ Height}$$

Since you already know the jump height and push height, you can calculate the JM. Then you can use the JM to calculate other aspects of the jump.

Example:

Jump Height = 1m

Push Height = 0.33m

JM = Jump Height/Push Height = 3

Jump Magnification and Acceleration

Jump Magnification is in fact an exact ratio that tells one how much the character has to accelerate against gravity to get into the air. The JM, besides being the ratio of jump-to-push vertical height and time, is also the ratio of push-to-jump vertical acceleration. Opposite the other ratios: while a longer jump time means a shorter push time, a higher jump acceleration means a much, much higher push acceleration. Knowing about this can help you make more informed decisions about your push timing.

To see how this works, let's look at the formula for JM and relate it to acceleration:

Jump Time Jump Height

$$JM = \frac{Jump\ Time}{Push\ Time} = \frac{Jump\ Height}{Push\ Height} = \frac{Push\ Acceleration}{Jump\ Acceleration}$$

The magnitude of jump acceleration is always equal to gravitational acceleration, with deceleration as the character rises and acceleration as it falls.

$$JH = \frac{Push\ Acceleration}{Jump\ Acceleration} = \frac{Push\ Acceleration}{Gravitational\ Acceleration}$$

Your landing speed is the same as the velocity of any falling object, which you can easily calculate from the free fall time. Since acceleration due to gravity is 10m/sec2, this means that after one second a falling object is traveling at 10m/sec, after two seconds at 20m/sec, after three seconds at

30m/sec, and so on. Since takeoff speed is the same as landing speed, you need to get up to that same speed when taking off for a jump. If your landing speed is 10m/sec, then during your takeoff you need to get up to a speed of 10m/sec in that little bit of push time.

The general formula for calculating the velocity of an accelerating object is: Velocity = Acceleration * Time

Physics shorthand: v = at

Let's relate this back to our jump. If the landing velocity is the same as the push velocity, we know that:

v = Jump Acceleration * Jump Time

So . . .

Jump Acceleration * Jump Time = Push Acceleration * Push Time

Moving things around with a bit of algebra, we arrive at this equation:

$$\frac{Jump\,Time}{Push\,Time} = \frac{Push\,Acceleration}{Jump\,Acceleration\,(Gravity)}$$

Look, it's the JM! And it's equal to the ratio of the push acceleration to gravity. Increase your jump time, and the push acceleration goes up. Decrease your push time, and the push acceleration goes up. Distance (or in this case, jump or push height) is also related to velocity: Distance = Average Velocity * Time

Physics shorthand:

d = vt

With some algebra, we make this into yet another formula for the average velocity:

v = d/t

Because the average velocity is the same for both the push and jump, we can say that d/t is the same for both jump and **push:**

Jump Height/Jump Time = Push Height/Push Time

And with a little more algebra:

$$\frac{Jump\,Height}{Push\,Time} = \frac{Push\,Height}{Push\,Time}$$

Push Time

The JM also gives you the ratio of the jump time to the push time.

JM = Jump Time/Push Time

Working a little algebra, we can express the equation in a way that directly calculates the push time:

Push Time = Jump Time/JM

Example:

JM = 3

Jump Time: 15 frames

Push Time = 15/3 = 5 frames

Landing

The forces on landing are similar to takeoff. If the landing has faster timing, the forces will be larger than for a longer timing.

Stop Time

The stop height is often a bit larger than the push height, but the timing ofthe push and stop are the same in the sense that the CG moves the same distance per frame in the push and stop. If the stop height is larger than the push height, you'll just need more frames for the stop than the push.

Push Height/Push Frames = Stop Height/Stop Frames

This can also be expressed as:

Push Height/Push Time = Stop Distance/Stop Time

You can also flip everything over and express it as:

Push Time/Push Height = Stop Time/Stop Distance

Using algebra, we can get the following equation for stop time:

Stop Time = (Push Time * Stop Distance)/Push Height

Example:

Push Time: 5 frames

Push Height: 0.4m

Stop Height: 0.5m

Stop Time = (5 * 0.5) / 0.4 = 6 frames

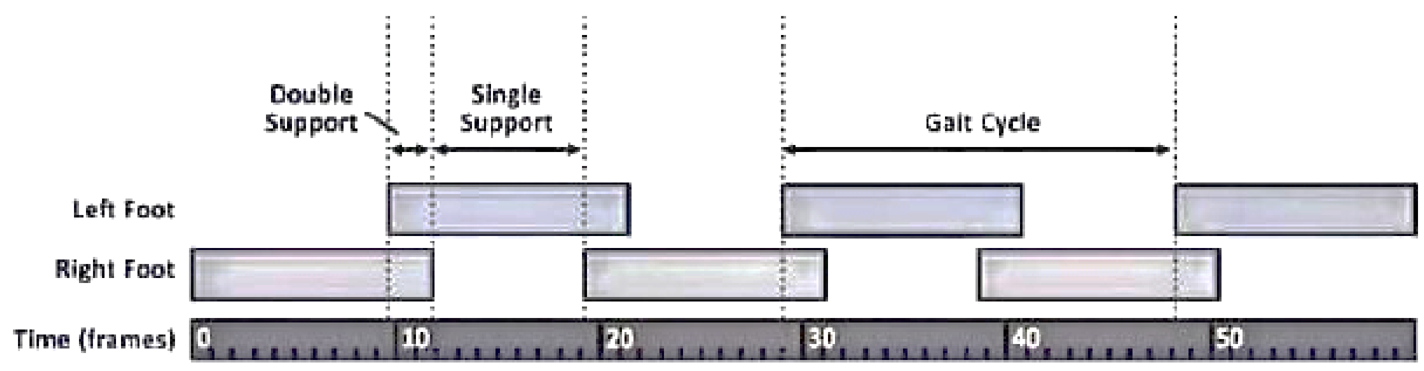
Walking

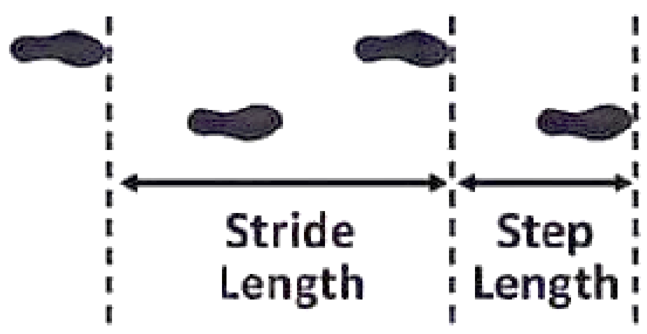
Walks feature all the basics of mechanics while including personality. The ability to animate walk cycles is one of the most important skills a character animator needs to master.

Strides and Steps

A step is one step with one foot. A stride is two steps, one with each foot. Stride length is the distance the character travels in a stride, measured from the same part of the foot. Step and stride length indicate lengthwise spacing for the feet during a walk.

Gait is the timing of the motion for each foot, including how long each foot is on the ground or in the air. During a walk, the number of feet the character has on the ground changes from one foot (single support) to two feet (double support) and then back to one foot. You can plot the time each foot is on the ground to see the single and double support times over time. A normal walking gait ranges from 1/ 3 to 2/3 of a second per step, with 1/2 second being average.





Walk Timing

Walking is sometimes called "controlled falling." Right after you move past the passing position, your body's center of gravity is no longer over your base of support, and you begin to tip. Your passing leg moves forward to stop the fall, creating your next step. Then the cycle begins again. The horizontal timing for between the four walk poses is not uniform. The CG slows in going from the contact to passing position, then slows out from passing to contact. The CG also rises and falls, rising to the highest position during passing and the lowest during contact. The head is in the highest position during passing.

Statistical Physics for Computing

Poisson Distribution

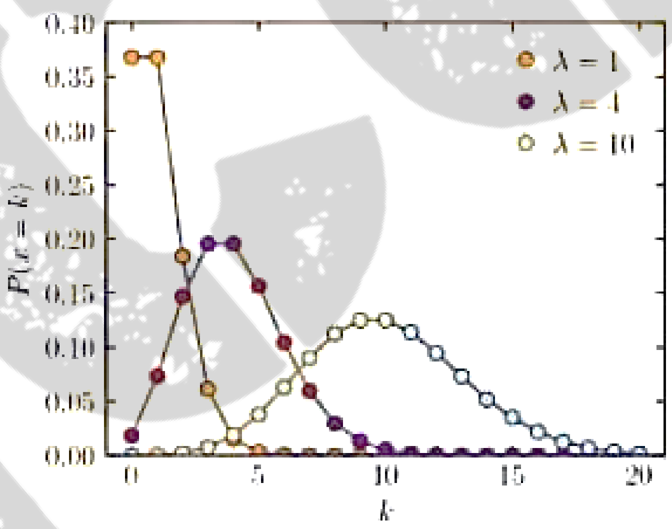
If the probability p is so small that the function has significant value only for very small k, then the distribution of events can be approximated by the Poisson Distribution.

Probability mass function

A discrete Radom variable *X* is said to have a Poisson distribution, with parameter , if it has a probability Mass Function given by

$$f(k;\lambda)=P(X=k)=\frac{\lambda^k e^{-\lambda}}{k!}$$

Here *k* is the number of occurrences, *e* is Euler's Number, ! is the factorial function. The positive real number *λ* is equal to the expected vlaue of *X* and also to its Variance. The Poisson distribution may be used in the design of experiments such as scattering experiments where a small number of events are seen.



Example of probability for Poisson distributions

On a particular river, overflow floods occur once every 100 years on average. Calculate the probability of *k* = 0, 1, 2, 3, 4, 5, or 6 overflow floods in a 100 year interval, assuming the Poisson model is appropriate.

Because the average event rate is one overflow flood per 100 years, *λ* = 1

$$f(k;\lambda)=P(X=k)=\frac{\lambda^k e^{-\lambda}}{k!}$$

P(k overflow floods in 100 years) = $\frac{\lambda^k e^{-\lambda}}{k!}=\frac{1^k e^{-1}}{k!}$

P(k=0 overflow floods in 100 years) = $\frac{\lambda^k e^{-\lambda}}{k!}=\frac{1^0 e^{-1}}{0!}=\frac{e^{-1}}{1}=0.368$

P(k=1 overflow floods in 100 years) = $\frac{\lambda^k e^{-\lambda}}{k!}=\frac{1^1 e^{-1}}{1!}=\frac{e^{-1}}{1}=0.368$

P(k=2 overflow floods in 100 years) = $\frac{\lambda^k e^{-\lambda}}{k!} = \frac{1^2 e^{-1}}{2!} = \frac{e^{-1}}{2} = 0.184$

Modeling the Probability for Proton Decay

The experimental search for Proton Decay was undertaken because of the implications of the Grand unification Theories. The lower bound for the lifetime is now projected to be on the order of $\tau = 10^{33}$ Years. The probability for observing a proton decay can be estimated from the nature of particle decay and the application of Poisson Statistics. The number of protons N can be modeled by the decay equation

$$N = N_0 e^{-\lambda t}$$

Here $\lambda = 1/t = 10^{-33}$/ year is the probability that any given proton will decay in a year. Since the decay constant λ is so small, the exponential can be represented by the first two terms of the Exponential Series.

$$e^{-\lambda t} = 1 - \lambda t \text{ , thus } N \approx N(1 - \lambda t)$$

For a small sample, the observation of a proton decay is infinitesmal, but suppose we consider the volume of protons represented by the Super Kameokande neutrino detector in Japan. The number of protons in the detector volume is reported by Ed Kearns of Boston University to be 7.5 x 10^{33}protons. For one year of observation, the number of expected proton decays is then

$$N - N_0 = N_0 \lambda t = (7.5 \times 10^{33}\ protons)(10^{-33}/\ year)(1\ year) = 7.5$$

About 40% of the area around the detector tank is covered by photo-detector tubes, and if we take that to be the nominal efficiency of detection, we expect about three observations of proton decay events per year based on a 10^{33}year lifetime.

So far, no convincing proton decay events have been seen. Poisson statistics provides a convenient means for assessing the implications of the absence of these observations. If we presume that $\lambda = 3$ observed decays per year is the mean, then the Poisson distribution function tells us that the probability for zero observations of a decay is

$$p(k) = \frac{\lambda^k e^{-\lambda}}{k!} \quad p(k) = \frac{3^0 e^{-3}}{0!} = 0.05$$

This low probability for a null result suggests that the proposed lifetime of 10^{33} years is too short. While this is not a realistic assessment of the probability of observations because there are a number of possible pathways for decay, it serves to illustrate in principle how even a non-observation can be used to refine a proposed lifetime.

Normal Distribution and Bell Curves

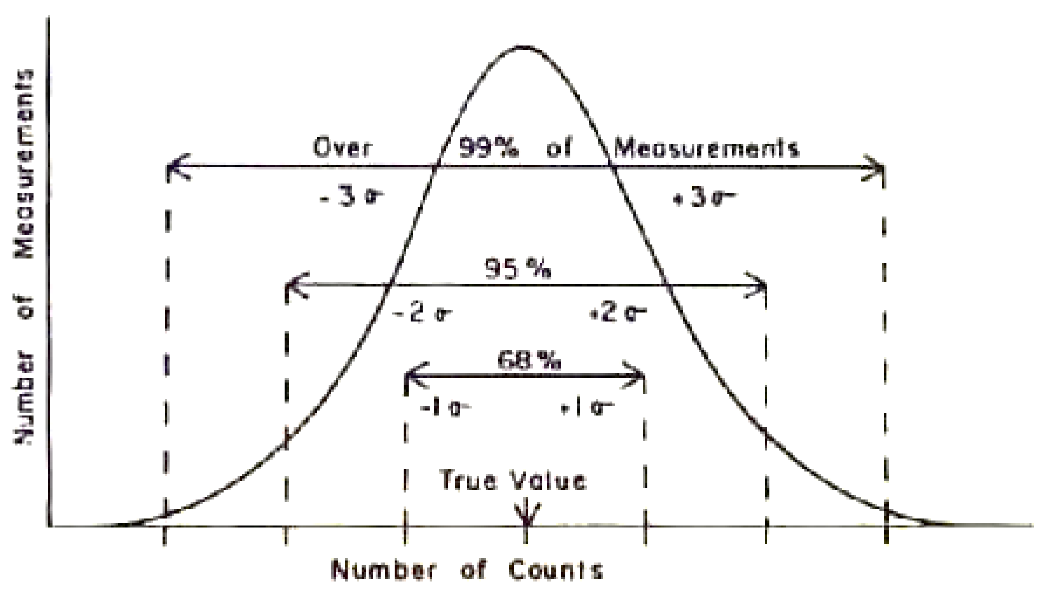
A bell curve is a common type of distribution for a variable, also known as the normal distribution. The term "bell curve" originates from the fact that the graph used to depict a Normal Distribution consists of a symmetrical bell-shaped curve.

The highest point on the curve, or the top of the bell, represents the most probable event in a series of data (its Mean, Mode and Median in this case), while all other possible occurrences are symmetrically distributed around the mean, creating a downward-sloping curve on each side of the peak. The width of the bell curve is described by its Standard Deviation.

The term "bell curve" is used to describe a graphical depiction of a normal probability distribution, whose underlying standard deviations from the mean create the curved bell shape. A standard deviation is a measurement used to quantify the variability of data dispersion, in a set of given values around the mean. The mean, in turn, refers to the average of all data points in the data set or sequence and will be found at the highest point on the bell curve.

Standard Deviations

The Standard Deviation is a measure of how spread out numbers are. 68% of values are within 1 standard



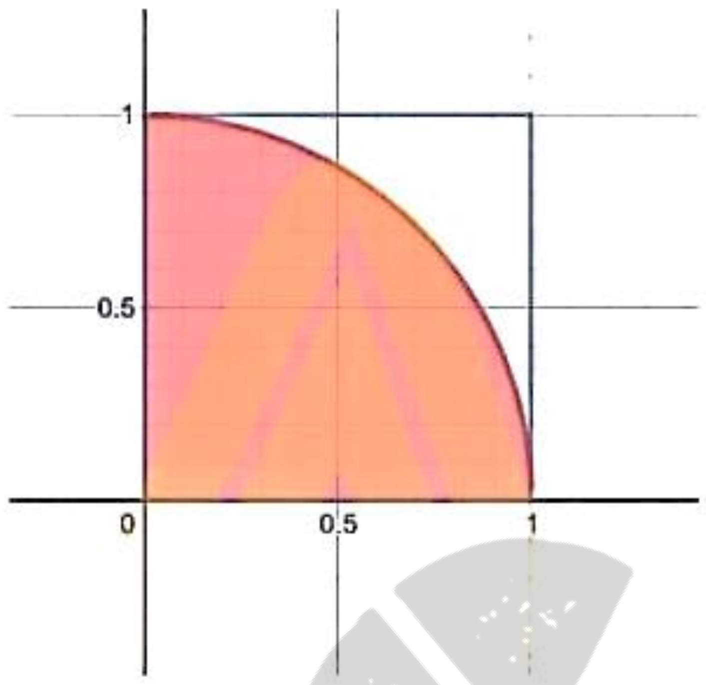
deviation of the mean. 95% of values are within 2 standard deviations of the mean. 99.7%of values are within 3 standard deviations of the mean

Monte-Carlo Method

Monte Carlo methods vary, but tend to follow a particular pattern:

1. Define a domain of possible inputs
2. Generate inputs randomly from a probability distribution over the domain
3. Perform a deterministic computation on the inputs
4. Aggregate the results

Monte Carlo method applied to approximating the value of π. For example, consider a quadrant inscribed in a unit square. Given that the ratio of their areas is $\pi/4$, the value of π /can be approximated using a Monte Carlo method:



1. Draw a square, then Inscribe a quadrant within it
2. Uniformly scatter a given number of points over the square
3. Count the number of points inside the quadrant, i.e. having a distance from the origin of < 1
4. The ratio of the inside-count and the total-sample-count is an estimate of the ratio of the two areas, $\pi/4$. Multiply the result by 4 to estimate π./

In this procedure the domain of inputs is the square that circumscribes the quadrant. We generate random inputs by scattering grains over the square then perform a computation on each input (test whether it falls within the quadrant). Aggregating the results yields our final result, the approximation of π.

There are two important considerations:

1. If the points are not uniformly distributed, then the approximation will be poor.
2. There are many points. The approximation is generally poor if only a few points are randomly placed in the whole square. On average, the approximation improves as more points are placed.

Uses of Monte Carlo methods require large amounts of random numbers, and their use benefited greatly from Pseudo random number generators, which were far quicker to use than the tables of random numbers that had been previously used for statistical sampling.