AST2000 - Part 1
Modelling a rocket Engine

Candidates: 76408 & 83886

(Dated: December 17, 2020)

In this study we are making a simplified model of a rocket engine by simulating the gas particles inside a very small section of the fuel tank. We made this section $10^{-12}m^2$, with 10000 H_2 particles inside at a temperature 3000K, and introduced a nozzle to see how many particles depart the tank. By doing this we found a change in momentum that results in a force exerted onto the shuttle. By summing up the change in momentum from all the small sections of the engine we find our engine's performance to reach 10^5N. This is then used to simulate a launch of our spaceshuttle, and we find that in order to reach an escape velocity of 13.4km/s we need to burn our engine for $\approx$ 6 minutes and spend ca. 1100kg of fuel.

I. INTRODUCTION

When a gas is heated and pressurized inside a gas tank, the particles of the gas will move with random velocities, this is called thermal velocity. We will use this thermal velocity to construct a rocket engine. When we introduce a nozzle, a hole in the tank, the particles will depart the tank with with the same thermal velocity - resulting in a change in momentum. This relates to a force exerted onto the shuttle by Newtons second and third law, meaning our fuel tank will thrust our rocket ship upwards into space to get us underway on our soon-to-be interplanetary voyage.

To simulate our engine, or rather the particles movement inside the tank, we will use numerical integration to simulate each particles motion using the thermal velocities in the gas. However, to do this we need to make some simplifications. Modern day computers are not powerful enough to perfectly simulate all particles inside a gas, so in this study we are going to simulate just a small section of the engine, with a set amount of particles and then use the superposition principle to sum up the effect from the entire engine.

All the physical principles and ideas needed to simulate the engine, as well as the actual launch, will be covered as we delve deeper into this study.

II. METHOD

Before we start our model we are going to make some major assumptions about the gas' behaviour inside the container to simplify our program, and minimize the computational load

1. The gas is pure H_2.

2. The particles do not interact with one another inside the gas chamber.

3. The gas is a Maxwell-Boltzmann ideal gas, meaning the thermal velocity follows a Maxwell-Boltzmann Gaussian distribution, and also both temperature and density of the gas is kept constant.

4. Collisions with chamber walls are elastic, meaning the kinetic energy and thus, its velocity is conserved.

5. Gravitational effects are negligible for the particles inside the gas.

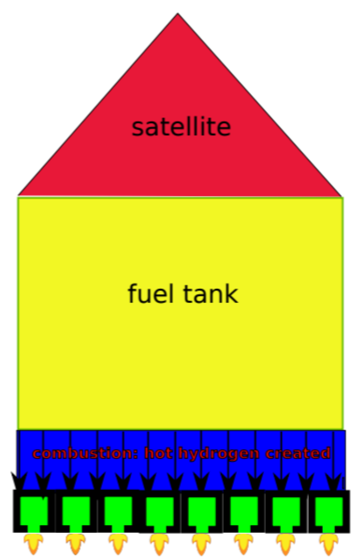

Figure 1: Simplified model of our spaceshuttle

This figure will give you a broad idea of what our planned shuttle will look like. We are going to model the fuel tank aswell as the nozzle, or combustion area on the bottom side of the fuel tank.

We will simulate all the particles inside the fuel tank individually using the differential equation for motion with constant velocity.

$$x(t) = x_0 + \frac{dx(t)}{dt}dt$$

which can be easily derived by approximating the derivative as a finite differential

$$v = \frac{dx}{dt} = \frac{x_{i+1} - x_i}{\Delta t}$$

$$x_{i+1} = x_i + \frac{dx_{i+1}}{dt}\Delta t \tag{1}$$

and we also know what the time derivative of position x, equals velocity v, meaning our differential equation becomes

$$x(t) = x_0 + v\,dt$$

The following figure will illustrate how the particles will behave inside the box which is LxLxL

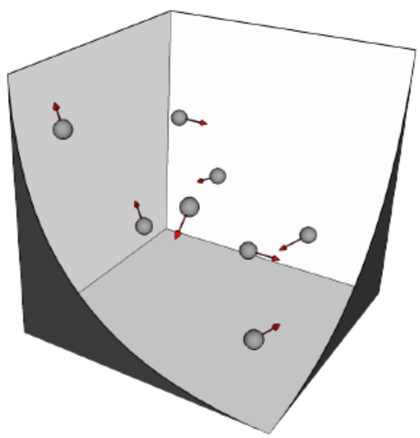

Figure 2: Energized particles inside gas chamber

As we can see from the illustration we are going to do the simulation in three dimensions, and as we know from basic calculus - to solve differential equations we need initial conditions. Meaning we need to know each particles initial position inside the container, as well as its initial velocity.
To find the velocity and position of particle i we do the following, using our previously mentioned assumptions:

- Draw a random number in the range (-L/2,L/2) for each of the coordinates x,y and z using the built-in pseudo-random-number-generator (PRNG) module random in Python to find a position where L is the sidelength of our container.

- Draw a random number in a Gaussian Distribution model using the parameters μ and σ from Maxwell-Boltzmann to find velocity in each dimension using the same built-in PRNG.

We are aware that Maxwell-Boltzmann distribution can be given by a Gaussian probability function using the parameters $\sigma = \frac{kT}{m}$ and $\mu = 0$, where μ is the mean value and naturally this is zero (we expect an equal amount of particles on the range [-vmax, vmax]) and σ the standard deviation of the curve, which is a measure of the width of the curve that is related to thermal velocity. Since we are assuming we have an ideal gas (follows from assumption 2 and 3), we can relate thermal velocity with temperature and mass by $v_t = \frac{kT}{m}$. See references for more details on ideal gases.

With our initial values, we can perform our numerical integration of the differential equation for motion using Forward Euler.
This numerical scheme is a more accurate twist on the regular Eulermethod, and it uses the same algorithm that we derived in (1) to solve this difference equation. (see references for a more detailed explanation of this numerical method)

Next up we are going to introduce a nozzle, which in principle will just be an area of our tank where particles can escape. We make it so that we can easily change the size of the nozzle, because before we start testing our simulations, we do not know the "best" size of the nozzle. Increasing, or decreasing the nozzlesize will result in more or fewer particles respectively, leaving the tank.

As per assumption (3), we want both temperature and density of the gas inside the container to stay constant. This means we have to compensate for each particle leaving the container, and we do this by adding a "new" particle at a pseudo-random position, using the method we previously mentioned, and give this particle a new velocity from the Gaussian distribution. We can imagine the "new" particles coming into the container as gas coming from the fuel tank into the combustion engine.
Furthermore, we need to take into account what happens when the particles collide with the containerwalls. We have assumed that these collisions are perfectly elastic, meaning that NO energy is lost from the collision, and thus the absolute value of the velocity must remain the same. The way to solve this is that we can check each particles position to see if it has reached outside of the box, i.e $x_i > L/2$ in any dimension, and if it has, we simply change direction in that dimension by multiplying the velocity in that direction, v_i, by (-1).

Now we have found a way to simulate our engine, and we will now look at how this engine will perform, and how it will help us ascend into space.
We are tracking each particle that leaves the engine, meaning we have a measurement of each particles velocity in every direction. Since we want our rocket to propel upwards, the only direction we need worry about is the z-component (you might think that it could be interesting to consider the other directions and if this

will cause of rocket to flip, but as you may remember the velocity follows a Gaussian distribution and therefore particles with velocity in x- and y-directions will cancel eachother out).

And we also know that each particle will have some form of momentum, since they have both mass and velocity, and momentum is given by

$$\mathbf{p} = m\mathbf{v}$$

Using this, we can rewrite Newtons 2nd law

$$\sum \mathbf{F} = m\mathbf{a} = m\frac{d\mathbf{v}}{dt} = \frac{d\mathbf{p}}{dt}$$

if we assume that the particles mass is time-independent. (Which it in most cases is). This shows that if we have experience a change in momentum, we will experience a force! This is the grand idea behind our rocket engine, we will "shove" particles out of our container with a force given from the change in momentum from the departing particles, and by Newtons third law, a force A always has an opposite counterforce B, we realize that the force exerted *from* our rocket onto the escaped particle, will equal to a force exerted *by* the particles onto our rocket.

This means that the change in momentum from our departing particles will exert a force that will thrust our rocket upwards! Not only that, we will also have a good measurement of exactly how much force is exerted, as well as how much fuel is being used while doing so. (since we measure both velocity and mass)

Now that we have a model of our rocket engine, and a way to measure force exerted onto the rocket from our fuel, we will want to test the engines performance.

In our previous simulations we used a very small container as an engine, and to actually produce the main rocket engine, we just superposition ALOT of these small engines, since we know the force output and fuel usage of each of the small ones.
With this we can compute the acceleration of our rocket, by using the rocket equation

$$\sum F_{ext} = \frac{d\mathbf{p}}{dt} = m\mathbf{a} + \mathbf{v}\frac{dm}{dt}$$

since momentum $\mathbf{p} = m\mathbf{v}$, and $\mathbf{v}\frac{dm}{dt}$ is the "force" exerted from the change in mass, i.e the thrust from our engine.
In this equation, $\mathbf{v}$ will be the relative velocity of the mass exerted by our rocket, compared to our rockets velocity. This means we have all the components in our equation, since we already have a measurement of the particles leaving our "box", we take the mean value of all the particles velocity and use this in our equation.
Now we have the acceleration given by

$$\mathbf{a} = \frac{(\sum F - \mathbf{v}_{rel}\frac{dm}{dt})}{m}$$

In this equation, $\sum F = F_G$ since gravity is the only external force affecting our shuttle (we are ignoring air resistance) and $\mathbf{v}_{rel}\frac{dm}{dt} = F =$ the total thrust from our engine and m is the current mass of the rocket (this changes as more and more fuel is spent boosting).

We know that the gravitational force will weaken as we ascend into the atmosphere as the distance, r, to the center of mass (planet center) will increase, and so we need to take this into account in our computations.
This force is given by Newtons Gravitational Law

$$\mathbf{F}_G = -G\frac{mM}{r^2}\hat{r}$$

where G is a gravitational constant, m the mass of our shuttle, M is the mass of the planet and r the distance from shuttle to planet center.

The thrust given from our engine we presume to be constant throughout our launch, since we keep both temperature and density constant, so the only variable we'll need is the distance, r, from the planets core.

With this we can easily calculate fuel consumed to reach a given velocity by computing the time spent accelerating to this given velocity, and using our measurement of fuel loss per timestep to find the total loss of fuel during the rocket burn. We can also implement and measure distance travelled with each timestep, which will give us a way to figure out how far our rocket has travelled during its launch. We define the launch as finished when we reach escape velocity.

Escape velocity is the velocity we need in order for our rocketshuttle to have a kinetic energy equal to, or large than the gravitational potential energy from the planet, meaning that we will no longer get pulled back into the planet if we stop accelerating. This can be put up as an equation

$$E_k \geq E_p$$
$$\frac{1}{2}mv_{esc}^2 \geq G\frac{mM}{r}$$
$$v_{esc} \geq \sqrt{\frac{2GM}{r}}$$

We will run our simulation of the launch until the velocity reaches this value, and at that point we will note how much fuel has been used, and our shuttles position relative to our homeplanet (x,y), and then convert it into the suns reference frame by doing a change of coordinatesystem

$$x_{new} = x_{old} + x_{planet} \quad (2)$$
$$y_{new} = y_{old} + y_{planet} \quad (3)$$

which is found by vectoraddition, since $\mathbf{x}_{old}$ points from planet to shuttle, and $\mathbf{x}_{planet}$ points from sun to planet and their vectorsum will point from sun to shuttle.

Before we initiate this launch, we remember that our planet has an rotational velocity that will give our rocket an initial velocity perpendicular to our launch direction (we are going to launch from the equator on our home planet, to maximize this effect since the velocity on the equator is higher than anywhere else on the planet surface assuming it spins around itself along the equator), meaning that we needn't accelerate as much to reach escape velocity since our velocity vector will already have a major component along the y-component as a result of the planets rotational velocity.

This value for rotational velocity, as well as all the values about our planet that we need in our computations are available to us through research done by colleagues at the ast2000 center for astrophysics.

III. RESULTS

We create a numerical program that can evaluate and find probability given by

$$P(a < x < b) = \int_a^b f(\mu, \sigma; x) dx$$

for a probability distribution function f. Where P will be the actual probability of finding x within the interval [a, b]. We can easily test if the integrator is working as intended by doing the integrals

$$P(-\sigma < x - \mu < \sigma)$$
$$P(-2\sigma < x - \mu < 2\sigma)$$
$$P(-3\sigma < x - \mu < 3\sigma)$$

where we find the expected values 68%, 95% and 99.7% respectively, and our implementation of the integrator is successful.
We'd like to see the Gaussian distribution of velocity for particles in our gas. To do this we use the two probability functions given by Maxwell-Boltzmann

$$P(v) = \sqrt{\frac{m}{2\pi kT}} e^{-\frac{1}{2}\frac{mv^2}{kT}}$$
$$P(v_x) = \left(\frac{m}{2\pi kT}\right)^{\frac{3}{2}} e^{-\frac{1}{2}\frac{mv_x^2}{kT}}$$

since a distribution given by Maxwell-Boltzmann is a Gaussian distribution with $\sigma = \frac{kT}{m}$.

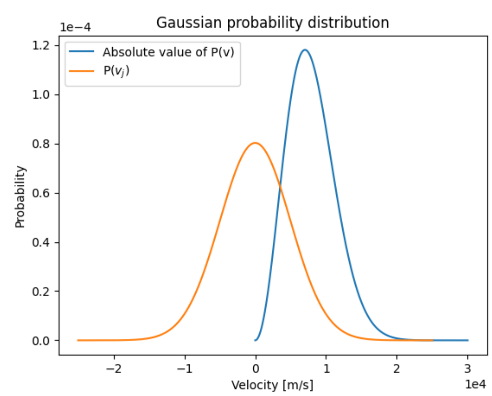

Figure 3: Gaussian distribution

We can easily implement these two functions into the same program we already have, and also plot their respective graphs (see figure (3)) against each other. We can use this program to check the probability of finding a velocity within a given interval by doing integral

$$P(a < v < b) = \int_a^b P(v_x) dx$$

numerically.
Testing this on the interval $v_x \in [-2.5, 2.5]\cdot 10^4$ and using the Simpson integration method we find the probability to be

$$P(v_x \in [-2.5, 2.5] \cdot 10^4) = 0.5$$

meaning theres a 50% chance of finding a v_x in the given interval. If we now multiply this with an arbitrary integer, say $N = 10^5$, we can find the amount of the N particles that'll have a velocity within that given interval.

Now we want to perform our simulation. Using the program we've explained in the method section we can simulate our box with $N = 100$ particles with a temperature $T = 3000K$ We see that our programs

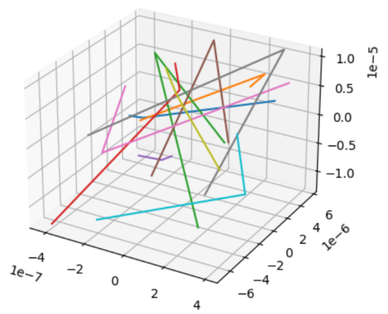

Figure 4: Plot of a few particles from our numerical simulation

behaves as intended, the particles "bounce" around inside our gas chamber. Now if we introduce a nozzle size of $0.8L$, and letting our program run for 1000 timesteps with steplength $dt = 10^{-12}$ gives an amount of ≈ 6000 particles leaving the gaschamber.

This number could be made larger by increasing the temperature, amount of particles, or size of nozzle, but it shows us that our program seems to work as intended for the time being and we expect a large number like this to escape when our nozzle is almost the size of the bottomside of the container.

In addition, the timeframe we run the simulation will also have an effect on this result, naturally, as more particles will have more time to escape our box. From the known number of escaped particles, we are able to calculate the total momentum of the escaped particles in one time interval. As explained in the method section, we take z-axis velocity component for the particles velocity that has escaped through exhaust hole. The resulting change in momentum from departing particles will then be

$$\sum_i m v_{zi}$$

Where m is the mass of the particles and v_{zi} is the velocity component in the z-axis.

Note that every simulation run will give us a slightly different value for the total change in the momentum due to the initial values being randomly distributed following a Gaussian curve as previously explained, however this deviation will not have a major effect since we are going to use ALOT of these small engines and all the small deviations will cancel out (we expect an equal amount to be less or more, since we have Gaussian distribution).

Since we have a 16 m^2 spaceshuttle we can fit $N = \frac{16}{L^2}$, where L is the sidelength of our box, onto the bottom of our rocket.

We will now increase our simulation drastically by introducing 10^4 particles, and with a sidelength of $L = 10^{-6}m$ and an exhaust hole (nozzle) with sidelengths $L_e = 0.8 * L$ we will be receiving a thrust of magnitude 10^{-9} Newtons from only one delta box using the rewritten expression for Newtons 2nd law we derived in the method section. However, by multiplying this with the N number of boxes we can fit on the shuttle

$$N = \frac{16}{(10^{-6})^2} = 1.6 * 10^{13}$$

we can see that the total thrust for our engine will reach up to about 10^5 Newtons. By knowing the thrust and the mass change we can now find the acceleration for our rocket by using the algorithm (rocket equation) explained in the method section.
With this expression for acceleration we can again apply Forward Eulers method to solve differential equations for motion, but this time for the entire rocket, with the term for acceleration given to use from rocket equation (see method section), and we count the number of timesteps needed to reach escape velocity,

$$v_{esc} = \sqrt{\frac{2 * 6.67 * 10^{-11} * 1.03 * 10^{24}}{7.75 * 10^5}} \approx 13400 m/s$$

and also track the position relative to the planetary surface.

Doing so lets us know that we need approx. 6 minutes of engine burn to reach a velocity of 13.4 km/s. Furthermore, we find that our engine burns ca. 1100kg of fuel. After the launch we find our shuttle at the position [1.28154526e-01, 1.14510588e-04] in astronomical units, relative to our star at the origin.

IV. DISCUSSION

As we saw in our result section, we managed to produce a net thrust of a magnitude of 10 kN with our engine. This was enough for us to manage reaching escape velocity, and it is also not too far off from some of the real life rockets that NASA are using (see references). However, when we constructed the rocketengine we filled the entire underside of our shuttle with the small "boxes" we modelled as our engine, and this is not very realistic having the entire bottomside of our shuttle filled with combustion engines. But it makes sense for our model, as we could see from figure (1). Also the nozzlesize is rather big, and the main reason for this choice is trial and error. We did multiple simulations with different sizes of the nozzle, and found that 0.8L was the best one, it gave the best force per fuel used for our shuttle.

When we calculated our escape velocity we used some basic physical principles, and we found 13.4km/s. Compared to the escape velocity from earths gravitational field, around 11.2km/s, this seems like a reasonable number.

V. CONCLUSION

We have made a simplified model for our engine. This model did however reproduce similar results as the real life rocket engines as we saw in the discussion section. We did not take into account air resistance when we simulated the rocket launch, and our engines performance may have been insufficient had we included this force as well.

All the results we got from AFTER the launch, i.e position and fuel usage, was given to us from our superb colleagues at the ast2000 facility! We were able to simulate a successful launch, but we were unsuccessful in matching their previous results (and they are rather accomplished astrophysicist so we can trust their simulations).

We were successful in our launch into orbit, but there are some flaws in the implementation of the actual launch. When we run our simulation, we find the spaceshuttle at a completely different position than the one we are expected to find it, our astronauts must've missed the correct exit!

ACKNOWLEDGMENTS

Special thanks to the head of AST2000, Frode Hansen for some great lecture notes and also, for aiding us in our study.

REFERENCES

- https://en.wikipedia.org/wiki/Euler_method Wikipedia - Detailed explanation of the different Euler numerical integration methods

- https://en.wikipedia.org/wiki/Thrust-to-weight_ratio Wikipedia - Thrust-to-weight ratio

- https://en.wikipedia.org/wiki/Ideal_gas Wikipedia - Ideal gases

- https://www.uio.no/studier/emner/matnat/astro/AST2000/h20/undervisningsmateriell/lecture_notes/part1a.pdf Lecture notes 1A

Appendix A: Derivation of FWHM (Full Width at Half Maximum

We see that the normal probability function f(x) given by

$$f(\mu,\sigma;x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{A1}$$

will have its maxima at μ , and thus we find f_{max} as

$$f_{max} = f(\mu) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{\mu-\mu}{\sigma})^2} = \frac{1}{\sqrt{2\pi}\sigma}$$

And so, if we want to find for which x, the function value $\frac{f_{max}}{2}$ occurs (Half maximum) we can setup the equation

$$\begin{aligned} f(x) &= \frac{f(\mu)}{2} \\ \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} &= \frac{1}{2}\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{\mu-\mu}{\sigma})^2} \\ e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} &= \frac{1}{2} \\ -\frac{1}{2}(\frac{x-\mu}{\sigma})^2 &= -\ln 2 \\ x &= \pm\sqrt{2\ln 2}\sigma + \mu \end{aligned}$$

We see what we would expect, that the x-value occurs twice, and finding FWHM means finding the difference between the two

$$FWHM = |x_1 - x_2| = |\sqrt{2\ln 2}\sigma + \mu - (-\sqrt{2\ln 2}\sigma + \mu)$$
$$FWHM = 2\sqrt{2\ln 2}\sigma. \ \blacksquare$$

Appendix B: Derivation of P = nkT

We will derive $P = nkT$ by solving $P = \frac{1}{3}\int_0^\infty pvn(p)dp$ where $n(p) = nP(p)$.
First we need to find an expression for the probability distribtion of momentum, $P(p)$ by using the relation $p = mv \rightarrow v = \frac{p}{m}$ we find

$$P(p) = (\frac{1}{2\pi mkT})^{\frac{3}{2}} e^{-\frac{1}{2}\frac{p^2}{mkT}} 4\pi p^2 dp$$

Inserting all of this into our original expression for P gives us

$$P = (\frac{1}{2\pi mkT})^{\frac{3}{2}} \frac{4\pi n}{3m} \int_0^\infty p^4 e^{-\frac{1}{2}\frac{p^2}{mkT}} dp$$

To make calculations simpler we introduce the constants

$$\begin{aligned} a &= (\frac{1}{2\pi mkT})^{\frac{3}{2}} \frac{4\pi n}{3m} \\ b &= \frac{1}{2mkT} \end{aligned}$$

And thus our integral becomes

$$a\int_0^\infty p^4 e^{-bp^2} dp$$

and by the substitution $u = bp^2$ we get

$$\frac{a}{b^{\frac{5}{2}}} \int_0^\infty u^{\frac{3}{2}} e^{-u} = \frac{a}{b^{\frac{5}{2}}} \frac{3}{4}\sqrt{\pi} = \frac{\left(\frac{1}{2\pi mkT}\right)^{\frac{3}{2}} \frac{4\pi n}{3m}}{\frac{1}{2mkT}^{\frac{5}{2}}} \frac{3\sqrt{\pi}}{4}$$

which, after reducing the fraction, becomes

$$P = nkT$$

which is what we wanted to derive $\blacksquare$

Appendix C: Derivation of $\langle E \rangle$

We will now be deriving The avarage value of velocity v by solving the integral

$$\int_0^\infty vP(v)dv$$

This deriving method is based on Maxwell-Boltzmans distribution. We know that

$$P(v) = \left[\tfrac{m}{2\pi kT}\right]^{3/2} e^{-\frac{mv^2}{2kT}} 4\pi v^2$$

If we include this expression into our integral, we end up with

$$\int_0^\infty v^3 \left[\tfrac{m}{2\pi kT}\right]^{3/2} e^{-\frac{mv^2}{2kT}} 4\pi dv$$

As for now, this integral seems to be a bit messy. We therefore gather all constants and make them into one single constant symbol.

$$C_1 = \left[\tfrac{m}{2\pi kT}\right]^{3/2} 4\pi$$

$$C_2 = \frac{m}{2kT}$$

Now that we have defined our constant, we can simplifie our integral

$$\int_0^\infty C_1 v^3 e^{-C_2 v^2} dv = C_1 \int_0^\infty v^3 e^{-C_2 v^2} dv$$

We substitute $C_2 v^2$

$$u = C_2 v^2$$

$$dx = \frac{du}{2C_2 v}$$

$$\frac{C_1}{2C_2} \int_0^\infty v^2 e^{-u} du$$

Because of our substitution, we see that $v^2 = \frac{u}{C_2}$

$$\frac{C_1}{2C_2^2}\int_0^\infty ue^{-u}du$$

We can look up to integral table and find out that

$$\int_0^\infty xe^{-x}dx = 1$$

We end up with the following expression

$$\frac{C_1}{2C_2^2} = \frac{1}{2(\frac{m}{2kT})^2} = C_1\left[\frac{(2kT)^2}{2m^2}\right]$$

$$4\pi\left[\frac{m}{2\pi kT}\right]^{3/2}\left[\frac{2^2k^2T^2}{2m^2}\right]$$

$$4\pi\left[\frac{1}{\pi}\right]^{3/2}\left[\frac{m}{2kT}\right]^{3/2}\left[\frac{1}{2}\right]\left[\frac{2kT}{m}\right]^2$$

We can see that we can algebraicly manipule exponents, because of

$$A^2 = A^{3/2}\cdot A^{1/2}$$

We can simplify our expression to

$$\frac{4\pi}{2}\left[\frac{1}{\pi}\right]^{3/2}\left[\frac{2kT}{m}\right]^2 = 2\left[\frac{2kT}{\pi m}\right]^{1/2}$$

Because of $2 = \sqrt{4}$ we can conclude that

$$\sqrt{\frac{8kT}{\pi m}}$$

■

Appendix D: Derivation of the average energy of molecule in an ideal gas

In order to derive this, we will be using the expression for kinetic energy

$$< E_k >=< \frac{mv^2}{2} >$$

We can write this as

$$\int_0^\infty \frac{mv^2}{2}P(v)dv$$

As for the $P(v)$, we will assume that the reader is introduce to the expression in previous appendix. We will also be using the same simplified constants (C_1 and C_2) as in the previous Appendix.

We can now rewrite our integral

$$C_1\int_0^\infty \frac{mv^4e^{-C_2v^2}}{2}dv = \frac{C_1m}{2}\int_0^\infty v^4e^{-C_2v^2}dv$$

We will be substituting

$$u = C_2v^2$$

giving

$$\frac{C_1m}{C_24}\int_0^\infty v^3e^{-u}du$$

We can see that

$$v^3 = \frac{u}{C_2}$$

and

$$v = \sqrt{\frac{u}{C_2}}$$

We implement these into our expression

$$\frac{C_1m}{C_24}\int_0^\infty \frac{u^{3/2}}{C_2^{3/2}}e^{-u}du$$

$$\frac{C_1m}{C_24C_2^{3/2}}\int_0^\infty u^{3/2}e^{-u}du$$

We use this integral in order to solve our expression

$$\int_0^\infty x^{3/2}e^{-x}dx = \frac{3}{4}\sqrt{\pi}$$

$$\frac{C_1m}{C_24C_2^{3/2}}\int_0^\infty u^{3/2}e^{-u}du = \frac{C_1m}{C_24C_2^{3/2}}\frac{3}{4}\sqrt{\pi}$$

$$\frac{C_13\sqrt{\pi}m}{16C_2C_2^{3/2}} = \frac{m\left[\frac{m^{3/2}}{(2\pi kT)^{3/2}}4\pi\right]3\sqrt{\pi}}{16\left[\frac{m}{2kT}\right]\left[\frac{m}{2kT}\right]^{3/2}}$$

$$\frac{24mkT}{16m} = \frac{12kT}{8} = \frac{3}{2}kT$$

■