PEAK OF FLIGHT NEWSLETTER

ISSUE 471 | June 12th, 2018

IN THIS ISSUE

What is the Optimum Engine Thrust Profile?

https://www.apogeerockets.com/Rocket_Kits/Skill_Level_4_Kits/Micro-Sentra_SRB

Apogee Components, Inc.

Your Source For Rocket Supplies That Will Take You To The "Peak-of-Flight"

4960 Northpark Drive Colorado Springs, Colorado 80918 USA

www.ApogeeRockets.com e-mail: orders@apogeerockets.com Phone: 719-535-9335 Fax: 719-534-9050

PEAK OF FLIGHT

What is the Optimum Engine Thrust Profile?

By Steve Ainsworth

Introduction

"The Goddard Problem" by Robert Goddard circa 1919, has been identified for over a century (do an internet search for papers). It is defined as follows:

Establish the optimal engine thrust profile for a rocket ascending vertically from the Earth's surface such that: 1) A given altitude is achieved with a given speed and a given payload, 2) The fuel expenditure is minimized, 3) Aerodynamic drag and the varying gravitation is considered, 4) The engine thrust is bounded.

The Goddard Problem includes altitude gained following engine burnout.

> "…the solution is shown to contain a finite number of such boosts in the sonic region of the rocket velocity, and to contain no coasting arcs except in the terminal stage" (Ref. 11)

The problem considered here is slightly different:

Establish the optimal engine thrust profile for a rocket ascending vertically from the Earth's surface such that: 1) A maximum altitude is achieved at engine burnout for a given mass of the rocket, 2) The fuel expenditure is given, 3) Aerodynamic drag and gravitational losses are considered, with gravity assumed to be constant, 4) The engine thrust is bounded.

This analysis ends with engine burnout.

The trajectory of the rocket depends on at least: 1) The mass of the rocket, 2) the thrust provided by the engine, 3) drag from the atmosphere, 4) overcoming gravity, 5) the launch angle, 6) wind effects.

For a given rocket, the variable to be optimized by this paper is the thrust profile provided by the engine.

Taking the simplifications discussed in the earlier *Peak Of Flight Newsletters* 455, 456, and 457, this paper develops an analysis method suitable for a computer program. The program finds the engine thrust curve that will provide the highest altitude for a given total impulse. To accomplish that, it first finds how velocity should vary to achieve the greatest altitude, then computes the engine thrust curve required to achieve that velocity.

Background

This chapter builds on the Apogee *Peak Of Flight Newsletters* 455, 456, and 457. Check them out for the detailed discussion of the equations used in this newsletter.

The equation notation convention provides the chapter number where the equation first appears and the equation number. Thus Equation 3.2 would be the second new equation listed in chapter three. Please note that equation and notation corrections regarding the previous *Peak Of Flight* articles are provided in this newsletter. A list of variable names and units are included in Appendix B.

For this analysis, we assume a zero degree launch angle with a perfectly vertical flight not deflected by winds aloft. This is a BIG deal and not easy to achieve with a static fin stabilized rocket.

Equation 4.2: Force from Drag: $FD = 0.5 * C_d * R_h * V^2 * A$: N
Equation 5.2: The Force due to gravity: Fg = Mrk * g = Mrk * 9.807: N

Multiplying the results of **Eqn5.2** by time step t we get The Impulse due to Gravity Ig:
Equation 5.2a: Ig = Mrk * g * t: N-s

Time Step

In order to calculate the impulse lost to gravity, we need to know the time spent under the gravitational influence. The approach here is to use a very small interval of time t, the increment of time, for the time step. That way we can use **Eqn5.2a** to calculate the impulse lost due to gravity for each time step.

$(h_f - h_i) = \Delta h$ is assigned by us, the smaller the value, the smaller the resulting t, and the more accurate the solution. The rocket velocity determines how long it takes to travel Δh m, and thus velocity and Δh provide the time step. Assuming that the rocket will accelerate uniformly from the velocity at the beginning of the time step to the optimum velocity at the end of the time step the average velocity during the time step (V_{av}) will just be one-half of the sum of the velocities at the beginning and the end of the time step.

$V_{av} = (V_i + V_f) / 2$

The time required is given by the change in altitude divided by velocity:

Continued on page 3

About this Newsletter
You can subscribe to receive this e-zine FREE at the Apogee Components website **www.ApogeeComponents.com**, or by clicking the link here Newsletter Sign-Up

Newsletter Staff
Writer: Steve Ainsworth
Layout/Cover Artist: Chris Duran
Proofreader: Will Franks

What is the Optimum Engine Thrust Profile?

Continued from page 2

$\Delta t = \Delta h(m) / (V_{av})(m/s)$; This is the duration of this time increment. Keep in mind that the duration of each time increment will not be the same, and will have to be computed independently. Also, the calculations provide a mathematical approximation and increases in accuracy with smaller time intervals.

Equation 6.1: $t = \Delta h/V$: s : Time Step Duration

Solving for V:
Equation 6.1a: $V= \Delta h/t$: m/s

Drag

We can plug V from **Eqn6.1a** into **Eqn4.2** and compute the resulting drag force in Newtons. Multiplying by the time step t we will get N-s of drag impulse losses.

Equation 5.1a: $ID = 0.5 * C_d * R_h *[\Delta h/t]^2 *A$: N of Force due to Drag. Or: $ID = 0.5 * C_d * R_h *[(\Delta h^2 *A / t$: N-s of Impulse due to Drag

It is convenient to substitute $Dconst = 0.5 * C_d * R_h *A$ so we get: **Equation 5.1b** $ID = Dconst *\Delta h^2/ t$: N-s

Dconst is only a constant over the time step t because it changes with each time step since C_d varies with Mach # (velocity, air density, and temperature) and R_h varies with altitude in accordance with a “standard atmosphere”.

Gravity

Another issue crops up. Since propellant is being consumed and exhausted, the mass of the rocket is decreasing so is not constant over the time step, and the mass of the rocket is required to solve Eqn5.2a. However, if the time step is very small, then the mass of the rocket throughout the time step is approximately equal to the mass of the rocket at the beginning of the time step. Again, in order for t to be very small, Δh must be very small. We will calculate the mass of the rocket at the beginning of the time step and assume it is constant for that time step, then update the mass calculation for the next time step.

Drag Plus Gravity

To calculate the total impulse loss due to gravity and due to drag, we can add **Eqn5.1a** to **Eqn5.2a**:
Equation 5.3: Total N-s loss = $ID + Ig = [Dconst *\Delta h^2 / t] + [Mrk * g * t]$

Keep in mind that:
- R_h varies with altitude but is assumed to be constant over time step t.
- Mrk varies as the propellant is burned but is assumed to be constant over time step t.
- C_d varies with velocity but is assumed to be constant over time step t.
- A is a constant (Reference area).
- g is a constant (Acceleration due to gravity)
- $t = \Delta h/V$ = time step duration
- V varies as the rocket ascends
- Δh is selected to be small in order to keep t small

Optimum Velocity

The total impulse loss due to gravity and drag together is provided by **Eqn5.3**. To get the most amount of useful impulse out of an engine we need to minimize that total impulse loss.

As velocity increases, the impulse loss due to gravity decreases. Faster is better.
As velocity increases, the impulse loss due to drag increases. Slower is better.
So, which is it?

Continued on page 4

What is the Optimum Engine Thrust Profile?

Continued from page 3

Figure 6-1 is an example graph with the y-axis as impulse loss and x-axis as velocity and one curve showing gravity loss, a second curve showing drag loss and a third curve which shows the sum of drag and gravity loss. It is clear from the plot that a minimum impulse loss exists for a specific velocity at a specific altitude. We will call this specific velocity the Optimum Velocity.

Optimum Velocity - Setting up a Computer Model

The program needs to be written to find the velocity that provides the minimum value for the total impulse loss from gravity plus drag, which will be the Optimum Velocity.

The gravity loss calculation requires knowing the velocity, time duration of the time step, and mass of the rocket including the propellant. That mass changes over the time step duration as propellant is burned to provide the optimum velocity at the end of the time step. The amount of propellant burned is dependent on the propellant used to achieve the optimum velocity. This is a circular calculation. To avoid the problem, we use the velocity and rocket mass at the beginning of the time step throughout the time step. This works as long as the time step is very small.

To calculate the drag loss, we need values for air density, drag coefficient, and velocity. If we assume that the air density and speed of sound is constant throughout the time step then we can calculate the drag loss for a range of velocities. We just need to be sure that the selected range includes the optimum velocity. We will use a large range of velocities.

The program will run the calculation for a number of velocities, and then find the velocity that provides the minimum total impulse loss from drag plus gravity.
From **Eqn5.3** we get:
ID = Dconst $*\Delta h^2$ / t: N-s
Ig = Mrk * g * t: N-s

We also have **Eqn6.1**: t = Δh/ V: seconds
Substituting for t
ID = Dconst $*\Delta h^2$/ [Δh/ V / V] =
Equation 6.2a: ID = Dconst$*\Delta h*$V: N-s
Equation 6.2b: Ig = Mrk*g*Δh/ V: N-s

Continued on page 5

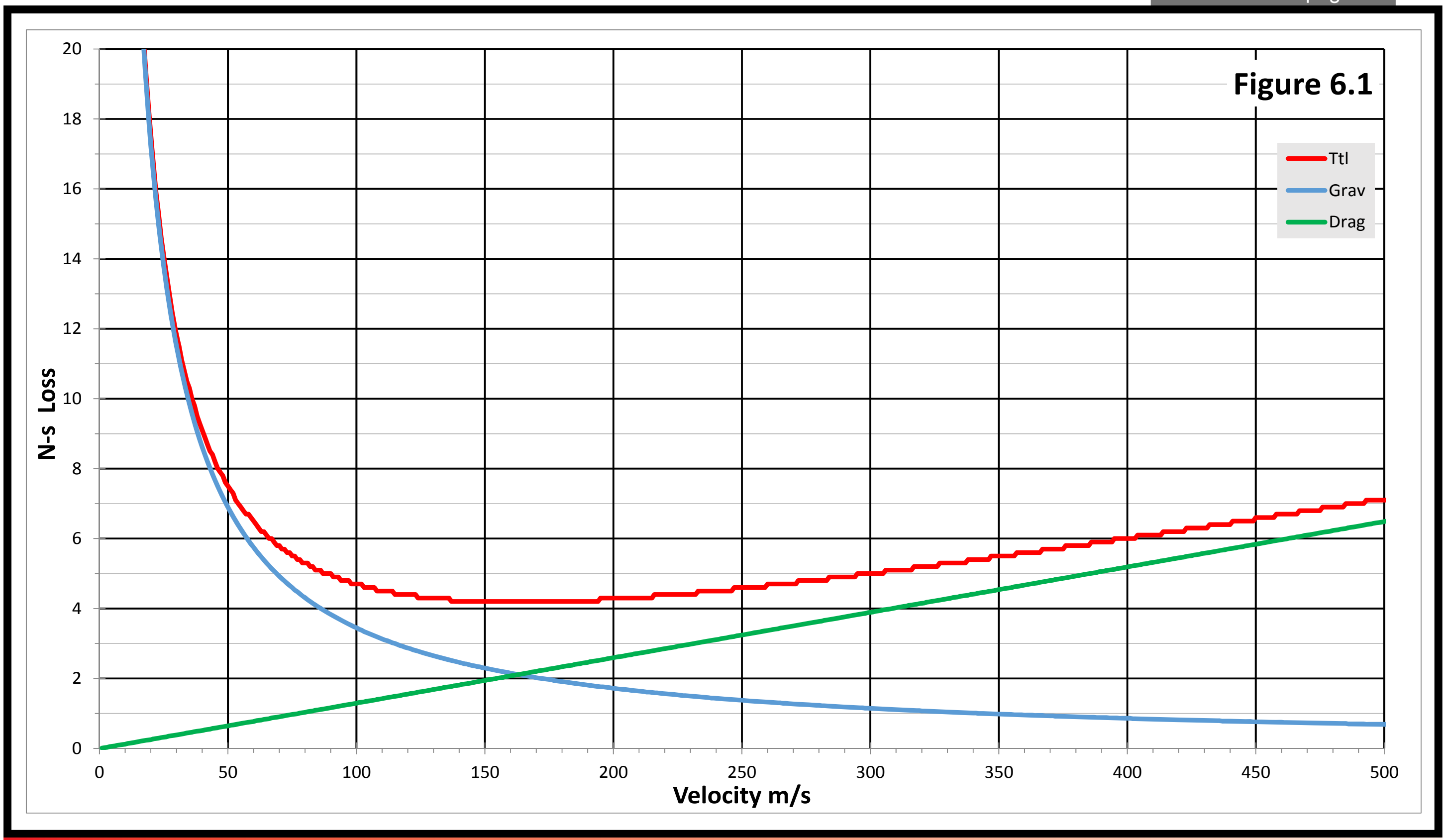

Figure 6.1: An example of Optimum Velocity

What is the Optimum Engine Thrust Profile?

Continued from page 4

Now we have both equations written as a function of Δh and velocity V. For the computer model we will select Δh and find the V that provides the minimum impulse loss. Using that "Optimum" velocity, and the selected Δh, we can calculate the resulting time step duration t.

The process then starts over with a new altitude initial: h_i. The new h_i provides the new R_h and the new speed of sound (Mach 1 speed). The new Mach 1 value combined with V_i provides the new C_d. Together these provide an updated Dconst.

Optimum Velocity - C_d as a function of Mach #

The Drag Coefficient changes with altitude and velocity. As it turns out, these variations are modeled fairly well with C_d as a function of Mach #. To obtain the C_d for the specific rocket as a function of Mach #, I ran the rocket model scenario using RockSim 9, and then exported a CSV (comma separated variable) file with C_d as the first column and Mach # as the second column using the RockSim export data tool. This provides the unique C_d as a function of Mach # for each rocket modeled.

Optimum Velocity - Polynomial Fit - Speed of Sound and Altitude

Based on a "Standard Atmosphere", I ran a polynomial fit curve for the speed of sound (Snd) as a function of altitude, which provided the following formula:

Equation 6.3: Snd = 2E-12*h^3 + 7E-08*h^2 - 0.004*h + 340.6: m/s

Optimum Velocity - Polynomial Fit - Air Density and Altitude

Based on a "Standard Atmosphere", I ran a polynomial fit curve for air density (R_h) as a function of altitude, which provided the following formula:

Equation 6.4: R_h = -4E-14*h^3 + 4E-09*h^2 + 1.218: kg/m^3

The program actually has Δh as an input, and then runs for velocities that increment 1 m/s per iteration over a range of velocities from 0 to 1500 m/s. For most rockets, the Optimum Velocity will fall within this range. If we keep Δh to around 10 m, then the time step duration is kept quite small. However, there are very many computations to run!

The output of part one of the program is the Optimum Velocity for a number of altitudes as the rocket ascends. If Δh is 10m then we have a value for each 10m increment of altitude. Since the things that vary with altitude do not change much over a 10m increment, then using the values at the beginning altitude for the entire time step is a good approximation to the exact answer. Another variable that is approximated is the propellant burned which effects the mass of the rocket. **Figure 6-2**.

Continued on page 6

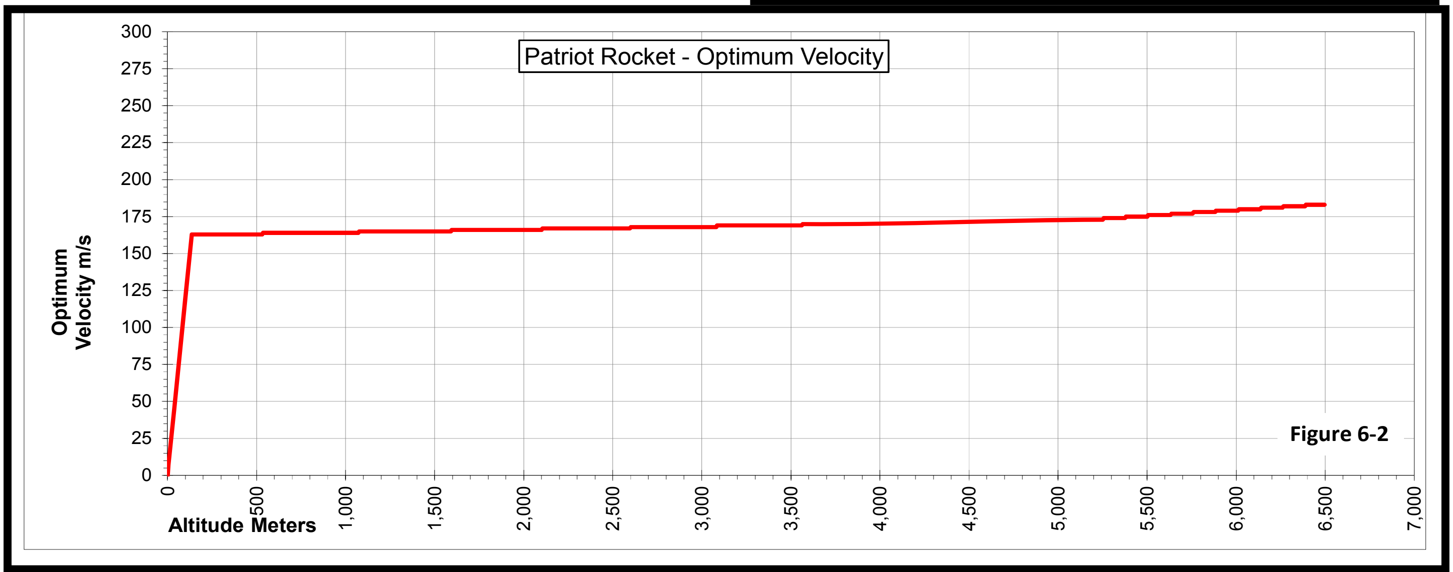

Figure 6.2: Optimum Velocity for the Patriot rocket

What is the Optimum Engine Thrust Profile?

Continued from page 5

Required Thrust

Each time step begins with the altitude, mass of the rocket, and velocity from the end of the previous time step. Using that data we find the optimum velocity and calculate the thrust Force necessary to produce the ΔV. In this way, we can build up a thrust curve as we calculate the desired velocity profile and include the reduction of mass from propellant burned. In order to have a model that provides acceptable accuracy, the time step t needs to be less than 0.05 seconds. A smaller Δh yields a smaller t.

Delta V (ΔV) - To Optimum Velocity

To get from one velocity to another, we need to accelerate the rocket. The velocity relationship to acceleration and time is given by:
Equation 5.4: $V_f = V_i + a * t$: m/s
Where:
- $(V_f\text{-}V_i) = \Delta V$ = Change in velocity: m/s
- V_i = velocity (m/s), velocity at the beginning of the time step
- V_f = velocity (m/s), velocity at the end of the time step, which is the calculated optimum velocity
- a = acceleration (m/s^2), acceleration required to go from V_i to V_f in time = t
- t = duration of the time step: s

Solving for acceleration we get:
Equation 5.4a: $a = (V_f\text{-}V_i)/t$ or $a = \Delta V/t$: m/s^2

Altitude is given by:
Equation 5.5: $h_f = h_i + (V_i * t) + 0.5*(a * t^2)$: m

Where:
- h_f = altitude at the end of the time step: m
- h_i = initial altitude: altitude at the beginning of the time step: m

Now we can compute the required acceleration for the time step that will provide the optimum velocity for the rocket at the altitude of the time step. With this in hand, we can calculate the required engine thrust N and impulse N-s for that time step:
Equation 5.6: F= Mrk * a + FD + Fg: N
Impulse = F*t: N-s

Combining **Eqn5.4a** and **Eqn5.6** we get:
Equation 5.7: F = Mrk * ΔV/t + FD + Fg: N

Where:
- F= Force in Newtons (kg-m/s^2) (of the engine)
- Impulse = N-s over the time step
- Mrk = overall mass in kg of the rocket at this time increment
- FD= Drag Loss: N
- Fg = Gravity Loss: N

Rocket Mass

We now calculate the mass of the rocket (Mrk) for each time increment. The rocket mass decreases each time step by the weight of propellant burned during that time step.
Equation 6.5: $Mrk_f = Mrk_i$ - Mpt : Mass final = Mass initial less propellant Mass consumed

Where:
- Mpt = mass of propellant lost during this time increment in kg.

For simplification, and so we do not have a circular equation, we will use:
Mrk_f from the previous time step = Mrk_i for the current time step during the entire time increment (Approximation)

We need to determine the mass of propellant burned (Mpt) during each time step that will provide the necessary thrust F to achieve the ΔV such that V_f is the optimum velocity.

To clarify, we need to know how much propellant will be burned in t seconds to produce the required thrust of F Newtons; how many kgs of propel-

Continued on page 7

What is the Optimum Engine Thrust Profile?

Continued from page 6

lant it takes to make F*t N-s of thrust. Since we know the desired V_f, we need to compute the kg of propellant required, which is Mpt.

Propellant Mass Flow Rate

The high powered rocketry hobby engine suppliers provide good engine data. The thrust curve displays the thrust (N) over the burn time of the motor. The motor designation provides the average thrust over the burn time. Therefore; a J350 has an average thrust of 350 N over its burn time, with a total impulse in the J range.

The motor datasheet also provides the total propellant weight (mass) Mp. Conveniently, RockSim's Engine Edit program provides the necessary engine data in an easy to get location.

The example rocket **Figure 6-3** is Brian Boers' Mad Cow Rocketry Patriot flying on an Aerotech L1120W.

The Aerotech L1120W has the following parameters:
- Mp = Propellant mass = 2.76 kg
- IT = Impulse total = 4922 N-s
- Isp = 182 s

Figure 6.3: Brian Boers' Patriot on the launch pad

- Burn Time = 5.01 s
- Peak Thrust = 1554.96 N
- Average thrust = 963 N

For the Patriot:
- Reference Diameter = 13.97 cm
- Total mass including propellant: 7.833 kg

We are looking for how much propellant mass we need to burn to provide the required Impulse:
Equation 6.6: Mp/IT = kg propellant per N-s
For the sample rocket: Mp/IT = 0.000645 kg/N-s

This is another approximation as the actual thrust does not vary linearly with kg of propellant consumed.

We have **Eqn5.7**: F = Mrk * ΔV/t + FD + Fg: and we have computed ΔV, t, FD, and Fg. We are using Mrk from the previous time step throughout this time step and as Mrki.

Now for this time step we have:
Mpt = [F*t N-s]* [Mp/IT kg/N-s] = kg propellant consumed in this time step.

Mrk for the end of this time step is then:
$Mrk_f = Mrk_i - Mpt$

Continued on page 8

What is the Optimum Engine Thrust Profile?

Continued from page 7

Getting the Rocket Moving

As mentioned several times in this discussion, the shorter the time step, the more accurate the result. That works great once the rocket is moving at optimum velocity. However, using the formulas above for the time between ignition and the rocket reaching optimum velocity results in a LARGE impulse at the beginning. And in this case, the shorter the time step, the larger the impulse needed to get the rocket from not moving to moving at optimum velocity in such a short time.

This issue is resolved by including a Peak Thrust (PeakF) N, for the engine, which is provided by the RockSim engine data. For a designer engine, you will need to determine this value. The program is then limited to using a thrust level that is at or below that maximum level.

Another artifact of this simplified analysis method is that the program tends to overshoot the Optimum Velocity by a small amount and subsequently wants the rocket to slow down in the next time step. The program is prevented from this instability by forcing it to accept the higher velocity. Thus the result could be a few m/s too high for a couple of time steps.

Putting it together

Figure 6-4 provides the calculated optimum thrust curve for the sample rocket. We could find a commercial motor with a thrust curve close to the optimum and use it for the flight or design a custom motor with the specific thrust curve that is optimum for our rocket. The optimum thrust curve doesn't look too exotic, a typical spike to get things moving, then a steady thrust to burnout.

Continued on page 9

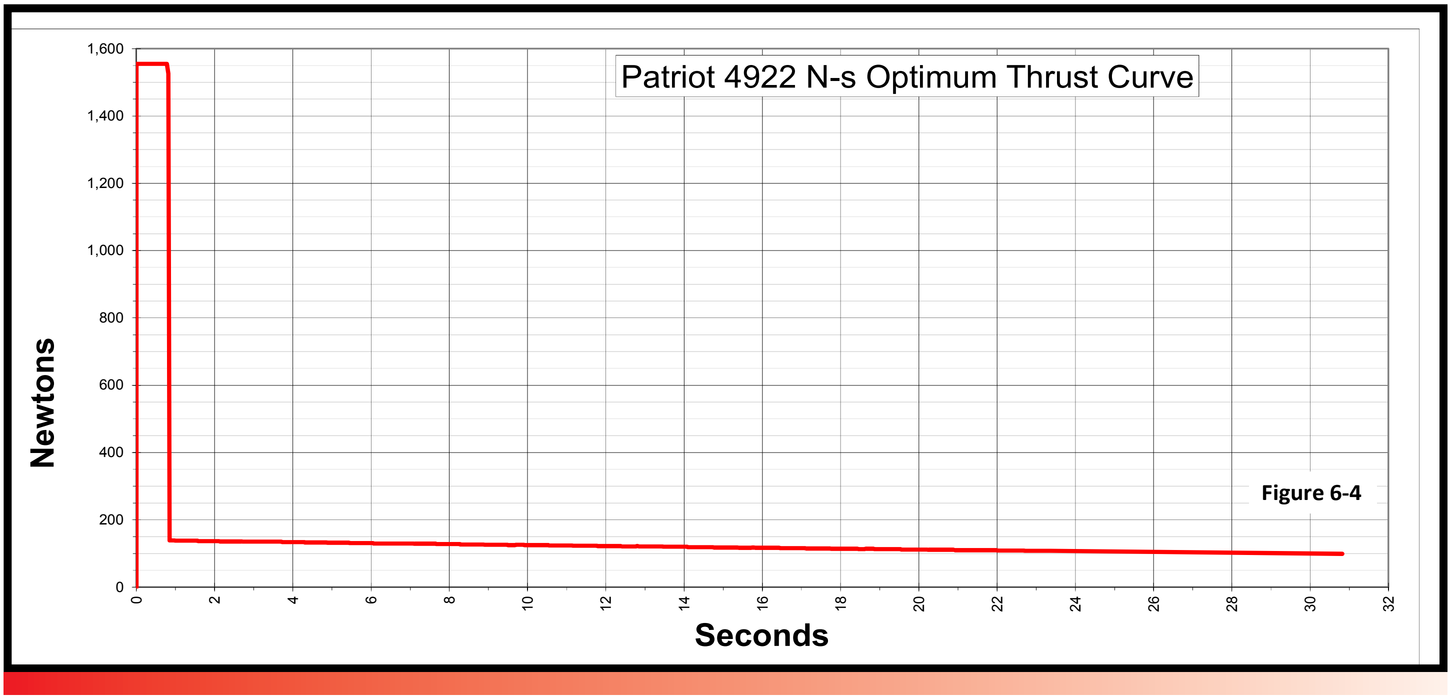

Figure 6.4: Calculated optimum thrust curve for sample rocket

What is the Optimum Engine Thrust Profile?

Continued from page 8

RockSim It

As a check, I entered this designer thrust curve into the flight simulation program RockSim v9 and "flew" the sample rocket with the designer engine to compare with the simulation using the actual engine. The results of the simulation (**Figure 6-5**) indicate that the redesigned thrust curve will provide an altitude increase from the original 2,109 m (6,919 ft) to 3,744 m (12,283 ft)! The prediction is for more than a 75% increase in the apogee altitude using the same size (total impulse) motor! The above results are for the C_d override value provided by Mad Cow Rocketry for the Patriot. If you use the RockSim computed C_d then the altitude goes from 3,498 m (11,476 ft) to 5,926 m (19,442 ft).

With RockSim, you can enter your own engine thrust curve using the EngEdit program provided, and tweak the thrust curve as needed to achieve the highest altitude for your rocket. When you do this, keep in mind that the above calculations are approximate for many reasons, and will only get you close to the optimum thrust curve. Several adjustments to the thrust curve may be required to get the most out of your total impulse. In addition, you may have to adjust the curve away from the optimum in order to have a thrust profile that is possible to construct given engine design limitations.

Enhanced Stabilization Systems (aka Vertical Trajectory Systems – VTS)

When you read the failure analysis of many of the high altitude attempts, often the problem is tied to either aerodynamic failure due to extreme velocities necessary to achieve a high altitude with a static fin guided flight or recovery failure due to excessive horizontal velocities at apogee. Both of these problems can be mitigated if the rocket can be kept going straight up.

Since beginning my quest for a VTS system in 1995, the R/C hobby industry has increased the accuracy of the over-the-counter stabilization systems. With improvements now driven by the R/C multi-rotor systems, stabilization accuracy is increasing rapidly.

With the recent advances, it may be possible to use a long burn motor to achieve high altitude without excessive rocket velocities in the lower atmosphere. It may also be possible to keep the rocket oriented properly to allow the proper deployment of the recovery system. If this piques your interest in the advantages of a vertical flight profile and VTS systems, check out reference 8 listed in **Appendix A**.

Author provided code & files that you can run to find the optimum thrust for your rocket projects. We put these free files on our web page for download here:
https://www.apogeerockets.com/Optimum-Thrust

Continued on page 10

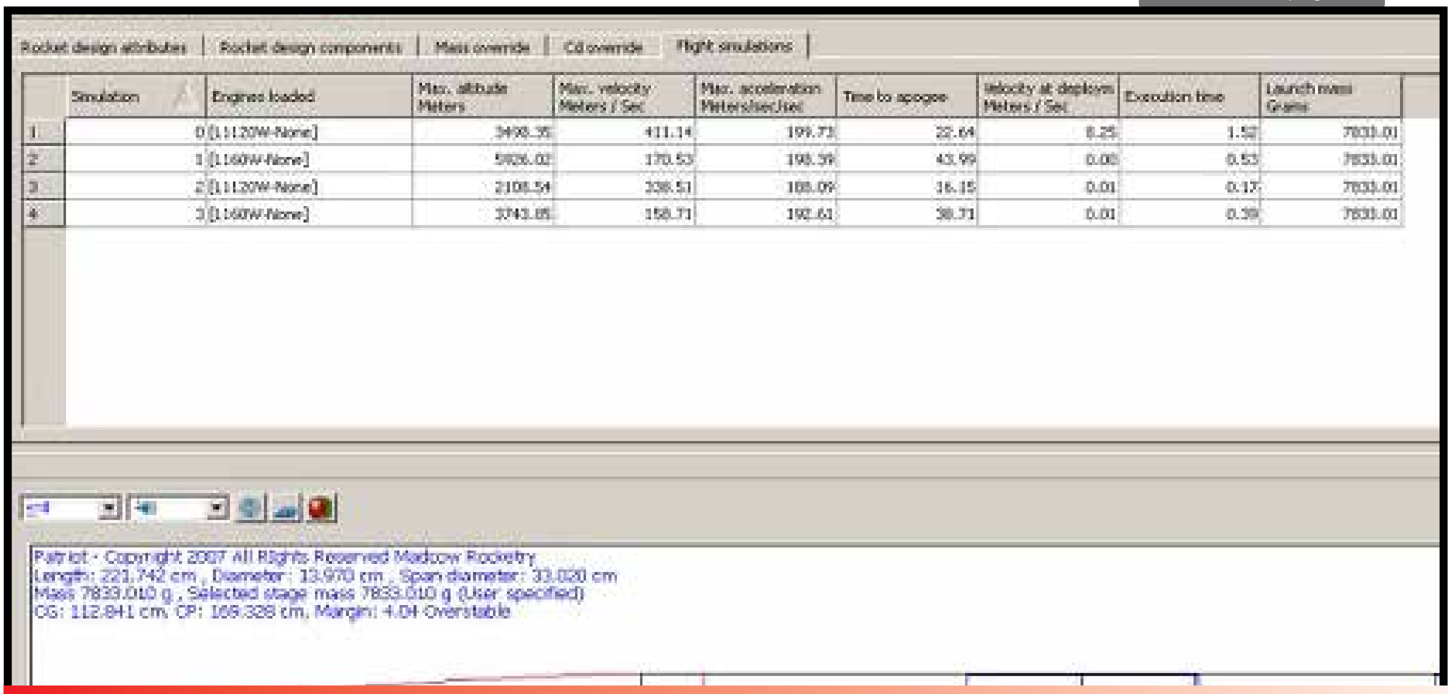

Figure 6.5: Screen shot of RockSim simulation for comparison

What is the Optimum Engine Thrust Profile?

Continued from page 9

Appendix A

References

Ref. 1: Mandell, Gordon K. Fundamentals of Dynamic Stability. NAR TR-201, 1973.
Ref. 2: Mandell, Gordon K. An Introduction to the Dynamics of Model Rocket Flight. From Topics in Advanced Model Rocketry. Published by MIT Press, 1973
Ref. 3: Ketchledge, David A. “Vertical Trajectory System and Dynamic Flight Mechanics for Model Rockets”. NAR SR 57237, HPR July/August, 1993.
Ref. 4: Goddard, Robert H. A Method of Reaching Extreme Altitudes. Smithsonian Institution, 1919
Ref. 5, 6, 7: Ainsworth, Steven. PeakOfFlight Newsletter #’s 255, 256, 257. APOGEE Rockets
Ref. 8: Ainsworth, Steven. Vertical Trajectory Systems. ARA Press, 2004. Available online
Ref. 9: Ainsworth, Steven. Optimal Altitude. ARA Press, 2011. Available online
Ref. 10: Caporaso, George. Elements of Trajectory Analysis. From Topics in Advanced Model Rocketry. Published by MIT Press, 1973
Ref. 11: Garfinkel, Boris. A Solution to the Goddard Problem. Ballistic Research laboratories Report No. 1194, Jan 1963

Appendix B

Variable names, units and Notation

A = Reference area of the airframe: m^2
a = acceleration: m/s^2
C_d = Drag Coefficient (Unitless). Varies with Mach number (approximately)
Dconst = 0.5*Cd*A = Drag Constant: m^2
dm = infinitesimal (almost zero) change in total propellant mass
dV = infinitesimal change in velocity of the vehicle
F = Force: Newtons: N: kg-m/s^2
g = gravity acceleration = 9.807 m/s^2
FD= Force due to Drag: N
Fg = Force due to Gravity: N
ID = Impulse due to Drag = FD * t
Ig = Impulse due to Gravity: N-s.
h = altitude: m
$\Delta h = h_f - h_i$: Change in altitude: m
h_i = altitude initial (meters), altitude at the beginning of the time step
h_f = altitude final (meters), altitude at the end of the time step
Isp = F/(ṁ * g) = Specific Impulse [kg-m/s^2]/[(kg/s)*(m/s^2)]: seconds
IT = Impulse total: N-s
kg = kilogram
m = meter
ṁ = propellant mass flow rate: kg/s
Mp = total mass of propellant: kg
Mpi = mass of propellant initial; mass at the beginning of the time step: kg
Mpt = mass of propellant burned during time step t in kg.
Mrk = Total Mass of the rocket including propellant: kg
Mrk_i = Total Mass of the rocket initial; mass at the beginning of the time step: kg
Mrk_f = Total Mass of the rocket final; mass at the end of the time step: kg
N = Newton = kg-m/s^2 [Force]
N-s = Newton-second = kg-m/s [Impulse]
PeakF = Maximum engine Thrust: N
R_h = Air Density: kg/m^2
s = second
t = time step duration: s
Ve = exhaust velocity of the propellant mass: m/s
V = Rocket Velocity: m/s
V_i = velocity initial (m/s); velocity at the beginning of the time step
V_f = velocity final (m/s); velocity at the end of the time step
$(V_f - V_i) = \Delta V$ = Change in velocity in m/s