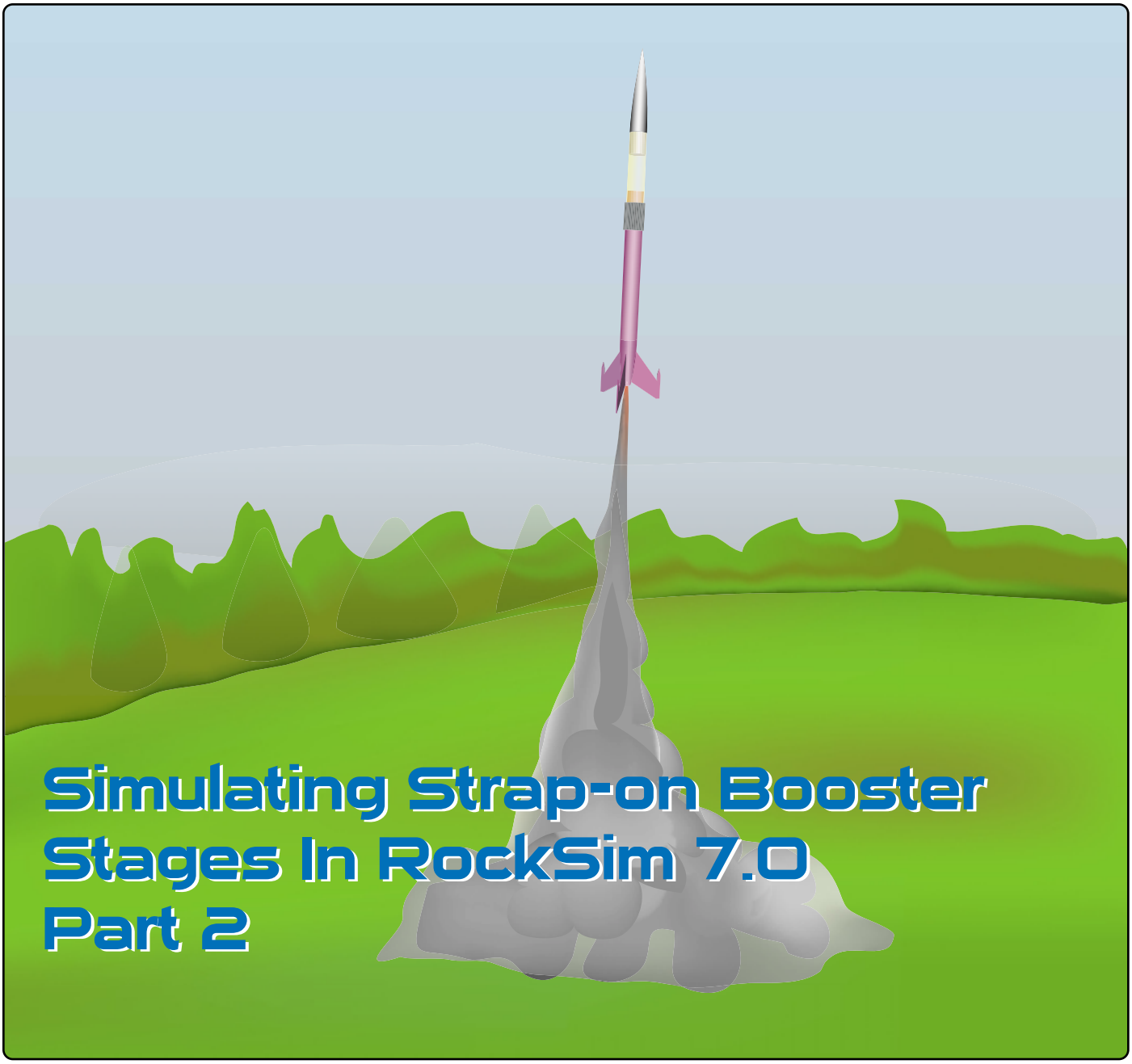
ISSUE 109, AUGUST 22, 2003

APOGEE PEAK OF FLIGHT NEWSLETTER



Simulating Strap-on Booster Stages In RockSim 7.0 Part 2

APOGEE COMPONENTS
1130 Elkton Drive, Suite A
Colorado Springs, CO 80907 USA
www.ApogeeRockets.com
orders @ApogeeRockets.com
phone 719-535-9335 fax 719-534-9050

ISSUE 108 - AUGUST 22, 2003

APOGEE

PEAK OF FLIGHT

Using RockSim To Simulate Strap-on Booster Stages - Part 2

By Tim Van Milligan

{ed. In the last issue, we started this series on parallel staging by looking at real-world examples of the technique. Then we took a look at what the thrust curve will look like for two different parallel staging cases. They differed by the time when the stages fell off the rocket. In this article, we'll discuss the basic concept of how we'll pull this off in RockSim.

To give you something to look forward to in the next installment, we won't actually walk through the steps yet. We'll save that for part 3 of this article. But if you can't wait for it, see the advertisement on page 6 on how to get the whole article right now, and even step-by-step videos on how to set the whole thing up in RockSim 7.0.

We'll now pick up where we left off in the last issue...}

Tricking RockSim

The first part of the answer comes from a trick I used about four years ago to create the Nike-Hercules design file. The Nike-Hercules is a conventional configuration 2-stage

Continued on page 3

Will Your Awesome New Rocket Design Turn Out To Be a Work-of-Art, Or a Piece-of-Garbage?

Your rocket design created in RockSim is very important to the success of the flight. But it is only the first part of the multi-step design process. If you assemble the individual parts wrong, it won't fly as high as RockSim predicts. If that happens, then what do you do?

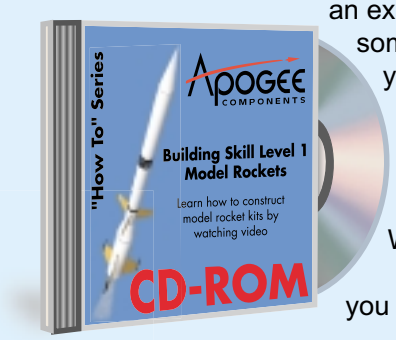
Most people think building a rocket is just cutting out parts and slapping them together with some glue. But let me ask you some questions:

- How do you make sure the parts fit together with no gaps?
- What type of glue should you use?
- How much glue is enough, and how much is "too much"?
- What happens if you use too much glue?
- How do you achieve a glass-smooth surface finish? ***How much difference does that make to the flight of the rocket?***

Over 70 Minutes of Video!

Building "quality" rockets is not something that comes naturally.

You can either learn through an expensive trial-and-error process (like I did), or you can learn by watching an expert. Look at it this way... Did you learn to tie your shoelaces by yourself, or did someone show you how? Isn't building rockets much more complicated than tying your shoes?



I have created a series of short videos, that will show you how to correctly assemble your rocket. When you watch these, it is like having me sitting right next to you. I'll show you how to correctly cut out your fins, how to make glue them on the rocket "straight-and-true," and how to get a flawless paint finish. When you are done, your rocket will fly straight, fast, and high.

Is paying $12.95 for your own private instructor worth it? I gurantee it is. And if you don't learn something new, I'll refund your money!

Format: CD-ROM (not DVD)
System Requirements: Windows or Macintosh
Requires: Adobe Acrobat Reader (v4 or newer) and Apple Computer's QuickTime player ("download" version included).

Order Today! Go To:
www.ApogeeRockets.com/skill_level_1_video.asp

Simulating Strap-on Booster Stages

Continued from page 2

rocket, where the lower stage has four motor tubes in a cluster configuration. To create the cluster, you have to put them "inside" a outer tube. But I didn't want the outer tube to affect the CG of the rocket, so I made it's wall thickness very thin. I think it was around 0.001 inches. It was so thin, that the weight is negligible.

So what I want to do in RockSim is to start the design of the booster stage with a very short tube, and one that has a very thin wall. This tube essentially becomes negligible to the design of the rocket. It adds practically zero weight.

Inside this tube, I'll put cluster engine mount tubes. These will actually be our "strap-on" motor tubes. We'll move the radial distance of these cluster tubes so that they appear to be outside the diameter of the core vehicle (the upper stage's body tube). This is a trick a lot of RockSim users have done to make outside pods.

Now all we have to do is move then forward, so their mass and CG location are in the correct position on the rocket. This last part is actually easy to do. You just assign a negative value for the "location" of the tubes.

Are you forgetting something?

"OK Tim," you say; "what about the nose cones that go on the front end of the strap-on boosters?"

I have to admit, that this part of the trick isn't going to give you a nice pretty picture in the 3D image. We have to make some compromises here. Which would you rather have, a pretty picture, or a accurate simulation result?

Nose Cones on Strap-on Boosters

To insure that we get a good simulation result, we have to ask ourselves some questions.

What would happen if I left the nose cones off?

Would it really matter?

To get down to the answer, we'll ask one more: *What affect do the nose cones have on the performance of the rocket?*

I'll answer this for you.

First, they add weight to the design. Nose cones have mass, which you can't ignore. Mass has a direct effect on performance.

Second, they add drag. As you expect, we can't ignore that either.

Finally, they change the CP location of the rocket. *Can we ignore this?* If we can't, can we find a way to compute it?

Let's continue by looking at each variable seperately.

Simulating the effect of weight is easy in RockSim. We can use a **mass object** to represent the nose cones on the strap-on boosters.

The drag effects can also be taken care of in RockSim v7.

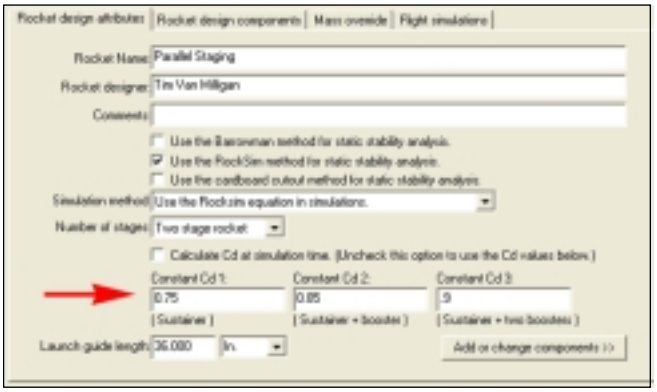


Figure 8: Each stage of a rocket can be given its own C_d value in RockSim v7. This can be used to assign a C_d value to strap-on boosters too.

In this new version, a feature that allows you to adjust the C_d of each individual stage was added. The reason we added it was for some schools participating in the Team America Rocketry Challenge requested it. They wanted to tweak the C_d of their designs. They discovered (good for them - they're learning!) that a single C_d value was not going to be accurate for both stages of their rocket. So in version 7, we added the ability to specify a separate C_d value for each stage of the rocket. See Figure 8.

It was a coincidence that we can use this feature allow you to make strap-on boosters.

What we'll need to do in our simulation is manually increase the C_d value of the booster stage to account for the nose cones on the tubes.

Continued on page 4

About this Newsletter

You can subscribe "FREE" to receive this e-zine at the Apogee Components web site (www.ApogeeRockets.com), or sending an email to: ezine@apogeerockets.com with "SUBSCRIBE" as the subject line of the message.

Simulating Strap-on Booster Stages

Continued from page 3

The only other variable is the CP. Does it change because there are nose cones on the strap-on boosters? Unfortunately: yes.

This complicates things; doesn't it?

This is where we'll have to cheat a little bit to make the simulations work.

To figure out how to get around this, take a look at the assumptions of the Barrowman Equations. If you study the them, you'll see that a nose cone always has a stabilizing effect on the rocket. In other words, when we add a nose cone on the strap-on boosters, they will help make the rocket more stable, not less stable.

Let's find a way to prove it.

Another item we can see that is alway stabilizing in the Barrowman equations is a "shoulder" type transition. So lets run a little test and move the location of the shoulder around on the rocket and see what effect it has.

We can easily test this in RockSim just by making up some simple finless rocket designs. Take a look at the CP position of the three sample designs shown in Figure 9 (Just look at the CP location, and ignore the CG position). You can see that no matter where we put the transition, it moves the CP rearward as compared to the rocket without the shoulder (the top image). Note that the further back we put the shoulder, the greater the effect it has at stabilizing the rocket.

What I wanted to show you from this little example is that the nose cones on the strap-on boosters are going to act much the same way as the shoulder transition section. They will actually stabilize the model.

It should be noted that the further back you put the nose cones (meaning shorter strap-on pods), the more stabilizing they will be. I would recommend that you do this to whatever extent possible.

But it is not all good news. You should recall that when you add strap-on boosters to the back of the rocket, you are also adding weight to the back end. From a stability point of view, this is also bad -- very very bad.

That is another reason to keep your strap-on boosters as small as possible: to shave weight off the back end.

I'd like to sum up this part of the discussion. While RockSim can't predict the CP effect of the strap-on booster's nose cones, we do know that they will have a stabilizing effect. So ignoring the CP requirement shouldn't concern us too much. If we design a "stable rocket" that doesn't have the nose cones on the strap-on boosters, we've reached the conclusion that the rocket will be even more stable once we actually put them on the real model.

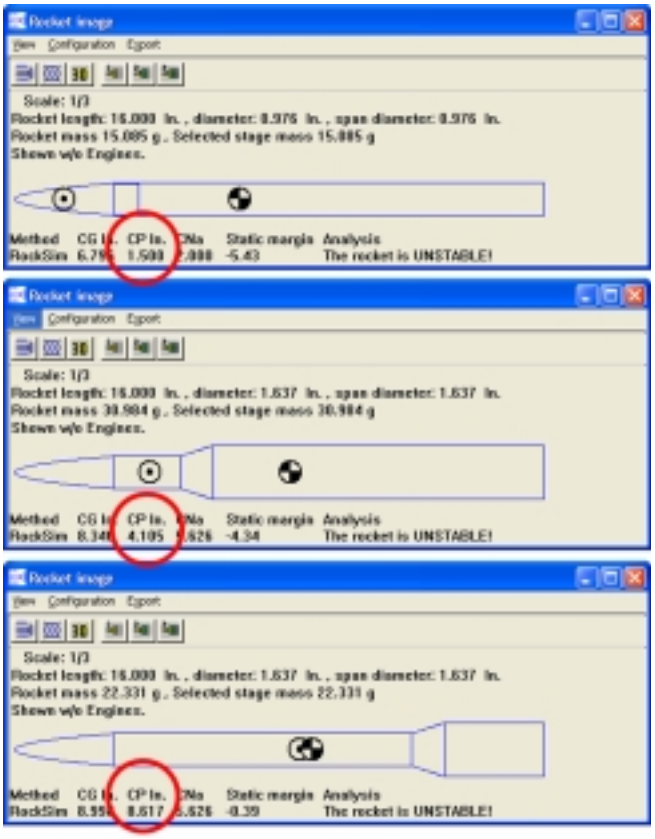


Figure 9: The nose cones on our strap-on pods will act a lot like the transition shown here. They will help stabilize the rocket.

The only time I would be concerned is if/when you are launching in high winds. This is where the rocket will be flying at a high angle-of-attack. At that point, the body tubes will start generating lift too. Again, this comes from the Barrowman Equations that state that tubes do not affect the CP position at all -- at low angles of attack. But when you have high wind, you're no longer at a low angle of attack. So the Barrowman assumptions are no longer valid.

My recommendation: don't fly strap-on rockets when it is windy; at least until we have a better understanding of how the CP will shift.

Design complete... Let's talk simulations!

At this point, we should have enough information to design the rocket in RockSim. But before I walk you through the steps, let's talk about simulations.

Continued on page 5

Simulating Strap-on Booster Stages

Continued from page 4

It is Case #2 that I want to talk about first. If you recall, we said that in this example, the upper stage will ignite as soon as the motors in the strap-on boosters burn out.

Simulating this will be easy. All we have to do is to treat the model like any ordinary two-stage rocket. The strap-on boosters will have motors that have zero second delays, while the upper stage will have a regular delay to it. For example, if we wanted to use Estes C6 motors in all the motors, we'd select C6-0 for the two motors in the strap-on boosters, and a C6-7 for the sustainer stage.

The only difference from a conventional two stage rocket is that we're telling Rocksim to change the C_d when the booster stage (our strap-on engine pods) drop off (see figure 8).

That was easy...

But now it gets toughter. What about Case #1?

I know what you're thinking. There is a Case #3 that we haven't talked about yet. It is when all the motors are lit on the pad, but the core vehicle has a really long burn motor. In other words, the strap-on pods are jetisoned, but the core continues to fire. It's just like the Delta II rocket we talked about before.

We'll get to case #3, but we have to talk about Case #1 first. Recall that in Case #1, all the motors in the model are the same type (we're using C6 motors in our example). They are all ignited on the pad, and they all burn out at the same time.

How do we get the motor in the core vehicle (our upper stage in the design file) to ignite at the same instant as the booster motors in the strap-on pods? That seems imposible in RockSim.

But it is possible to trick Rocksim.

Simulation Tricks For Strap-On Boosters

Forcing RockSim to ignite the core motor (which is in the sustainer stage), at the same instant as the motors in the strap-on boosters is going to require a little ingenuity.

In Apogee Technical Publication #14, I wrote about other tricks you could use to perform some complex simulations. http://www.apogeerockets.com/technical_publication_14.asp

That article was written a long time ago, and most of the tricks are no longer necessary. Because over the years, we've added features into RockSim, like unlimited cluster arrangements, that did away with the "trickery."

However, there is one little trick that is still has some usefulness — simulating boosted darts.

In a boosted dart, you have an unpowered top stage. Since it doesn't have a motor in it, there was no way to control when the recovery device would be deployed. We discovered that the trick is to make a phantom motor. This false motor needed to have zero mass, and a thrust curve that is practically nonexistent. By making the zero-thrust motor for the dart stage, we force Rocksim to keep the simulation going after the booster stage drops off.

This phantom motor isn't really necessary an more, since RockSim now has a way to keep the simulation going even if the dart stage doesn't contain a motor. This is done using the Flight Events and Simulation Controls feature of RockSim. If the Flight Event is set to deploy the streamer at apogee, and the Simulation Control is set to end at apogee, you don't need a motor in the top stage. You can find out how high the rocket will fly regardless.

The only time you'd still need the phantom motor in a boosted dart simulation is if you want to track the descent of the dart stage. I'll leave this excersize up to you to play with.

The Phantom Motor

The trick we're going to use in this situation is somewhat similar.

What we need is a modified (fake motor) to go in the booster stage that is ignited along with the strap-on booster motors. This motor will have thrust — and propellant mass — but no casing mass.

Since we're adding a new motor to the "booster stage" of the RockSim design file, we'll need a new engine mount tube added to the cluster. That is easy to do.

Instead of our booster stage having two motors mounts, it will be a three engine cluster.

Let's talk more about the modified booster stage motor first.

We want this modified booster motor to be ignited on the ground. It should have all the characteristics of the sustainer stage motor in the "real" rocket -- except for one thing. We need to change the mass of the motor to only represent the propellant.

In other words, we're taking the mass of the propellant -- which should be in the top stage -- and we're moving it down to the bottom stage. In our case, the bottom stage is the strap-on pods.

Additionally, we need to move the "thrust" of the top stage, and add it to the bottom stage. Why? Because the top stage in our case study has to ignited on the ground when the button is pushed.

Why couldn't we just put a unmodified motor into the bottom stage instead of creating a modified engine file?

Continued on page 6

Simulating Strap-on Booster Stages

Continued from page 5

Because the weight of the empty motor case needs to remain in the top stage after the motor burns out. If we used a unmodified motor in the bottom stage, when the strap-on pods fall off, it would take the mass of the empty case with it. We want that case's mass to stay with the top stage.

The Empty Casing

Just like in the real flight, after the booster pods are jettisoned, the core vehicle continues to coast upward. But remember, in real life, there will be a empty casing in that part of the rocket. We have to account for the mass of this motor case, because it could be significant to the coast performance.

There are two ways we could do it.

The simple way is to use a mass object in the upper stage. But you'll have to look that up for the motor you want to use. To find the information, you'll first open up the EngEdit program, and find the motor file you want. The case weight is the total weight minus the propellant weight.

Once you know this information, you just open up your RockSim design file, and drop a mass object in the rocket somewhere close to the back end of the rocket.

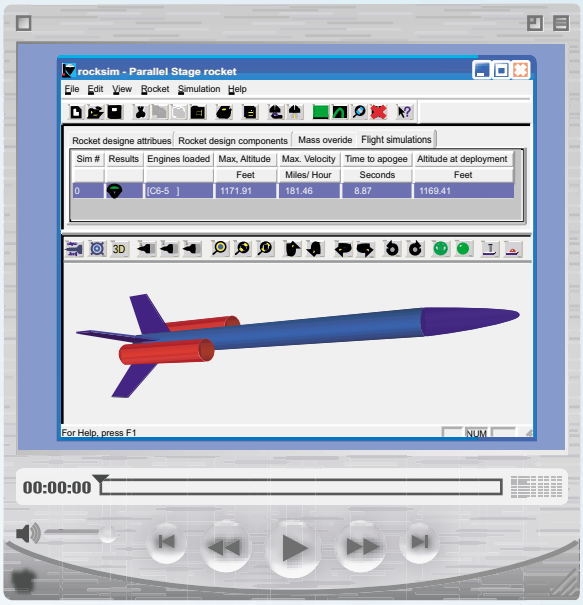
The other method is method is to make a fake motor (again

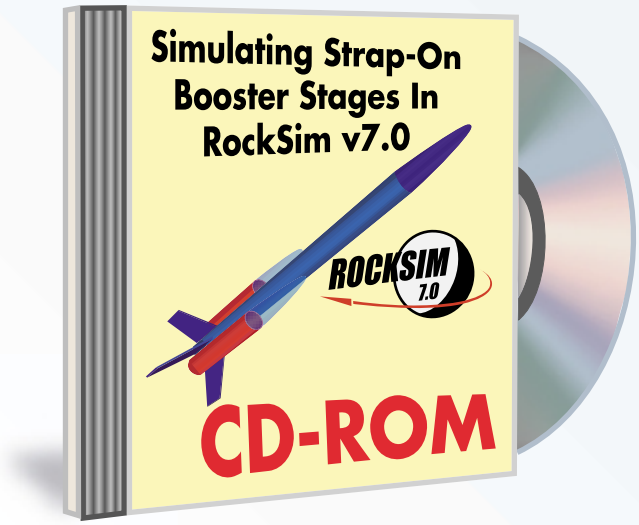
Continued on page 7

Discover RockSim Tricks & Tips As You Learn How To Simulate Strap-On Booster Stages!

Learning the techniques to simulate strap-on booster stages in RockSim offers me the opportunity to demonstrate to you some of the advaced features that I use every day. I can teach you how to make your RockSim simulations more accurate, and how to create your designs quicker.

This new CD-ROM has over 34 minutes of how-to instructional videos on it that will teach you: how to use the Engine Editor software, compiling the engine database, analyzing the simulation results, and lots of tricks and tips.





Learn At Your Own Pace By Watching Computer Videos

For just $10.95, you not only get the high quality how-to videos, you'll also get 11 additional articles that explain other advanced features of RockSim, and understanding rocket science. You'll be the expert that other novice modelers rely on for advice.

Order today. Just go to the Apogee Web site at:

www.ApogeeRockets.com/strap-on_boosters.asp

Simulating Strap-on Booster Stages

Continued from page 6

using EngEdit), and then load this motor into the rocket when you run your simulation. There are two advantages to doing this, even though it takes more time to set it up.

First, the case mass will be positioned in the right location inside the rocket. You won't have to approximate its location like you do with the mass object discussed above.

And second, if you have a motor (even a fake one) in the upper stage, RockSim will be able to track it all the way to touchdown. So you'll find where the upper stage will land. If you just use a mass object, the most data you can get from the simulation is the maximum altitude.

This fake upper stage motor will be just like our boosted-dart motor, except now it will have the weight of the casing. (A boosted dart motor has to have zero weight -- but in this situation, the phantom motor will have some mass to it). So our phantom motor needs some weight, but near-zero thrust.

What happens during the simulation?

Just so you understand what is going on, let's walk through how this will all work. Recall that in this case we are using the Estes C6 motor for all the motors in the rocket.

Strap-On Booster Stage - In our RockSim design file, this stage will consist of a cluster of three motors. Two C6-0 motors, and one *modified C6-0* motors that has only propellant weight.

Core Stage - Contains a single phantom motor. The purpose of this motor is to trick Rocksim to continuing the simulation after all the propellant in the lower stage is consumed. This motor only has case weight, and a very tiny thrust curve.

Note: It would be desireable to have zero thrust in the phantom upper-stage motor. But the requirements of the ".eng" file structure force us to to have some thrust. We'll get around that by making the thrust so small that it becomes negligible to the simulation.

Now in our simulation, when we push the launch button, the three motors in the booster stage ignite. They consume the entire propellant in the whole rocket. This satisfies the requirement that all three C6 motors in the rocket have the same thrust and burn out at the exact same instant.

After they burn out, the strap-on pods stage drops off. When it goes, it takes the entire weight of strap-on pods and their nose cones.

Also remember, we have given the booster stage it's own C_d value. So when the booster stage drops, the C_d changes to reflect just the upper stage (core vehicle). Everything is going as planned so far.

To continue with the simulation, our new phantom motor in the upper stage ignites. Since it's thrust curve is so short, it burns out nearly as soon as it ignites. Now the delay element starts burning, and the rocket coasts upward. The empty case from the motor is still inside the upper stage; because we gave the phantom motor some mass when we created it. Finally, the ejection charge is fire in the phantom motor, deploying the streamer. Since there is a motor in the top stage, we can let RockSim continue with the simulation and track it all the way until it touches down. When it does, we'll know the peak altitude, where staging occurs, what is the best delay to use, where the upper stage lands, and how long it takes for the rocket to complete its entire flight – which is what is the whole purpose of this exercise.

Coming Next Issue

In the next issue, we'll start walking step-by-step through the exact proceedure you'll follow to add parallel staging in RockSim. If you've ever wanted to know how to use the Engine Editor software, this will be your chance to find out how powerful and versatile it really is.

About the Author:

Tim Van Milligan is the owner of Apogee Components (http://www.apogeerockets.com) and the curator of the rocketry education web site: http://www.apogeerockets.com/education. He is also the author of the books: *"Model Rocket Design and Construction," "69 Simple Science Fair Projects with Model Rockets: Aeronautics"* and publisher of the FREE e-zine newsletter about model rockets. You can subscribe to the e-zine at the Apogee Components web site, or sending an email to: ezine@apogeerockets.com with "SUBSCRIBE" as the subject line of the message.

Team America Rocketry Challenge is back!

See the Apogee Components web site for tips and suggestions on improving your chances.

www.ApogeeRockets.com/team_america_challenge.asp