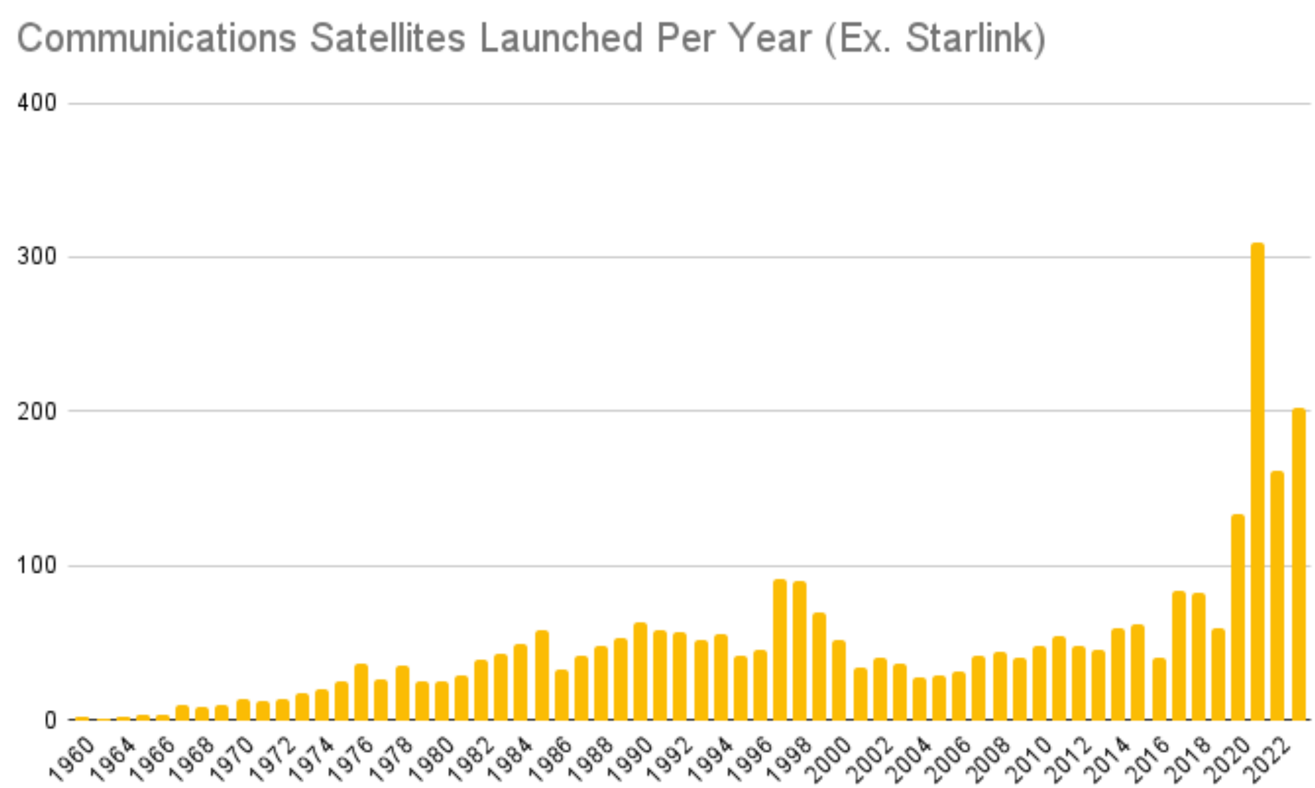
Christopher Kalitin Blog

Fun Charts That Visualize The History of Communicatons Satellites

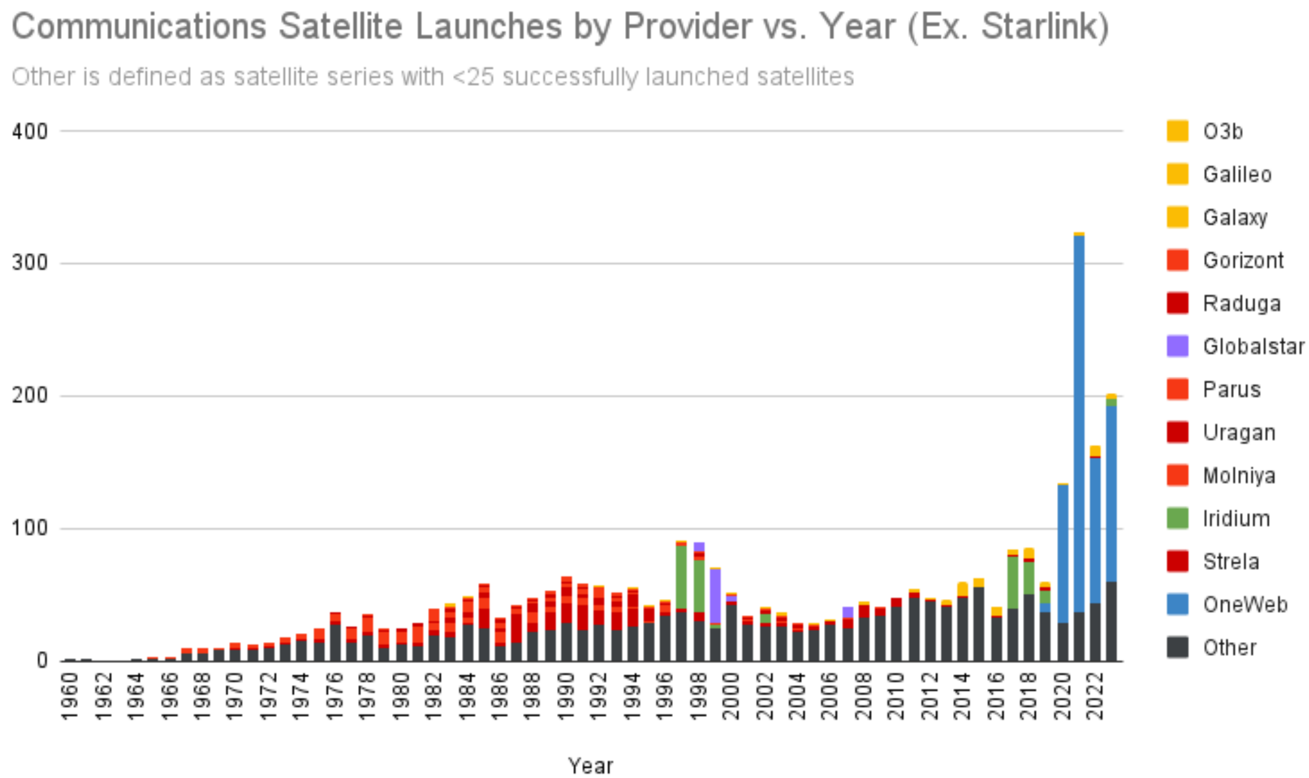
Jan 15, 2025 • Christopher Kalitin



This is the chart from the previous blog post.

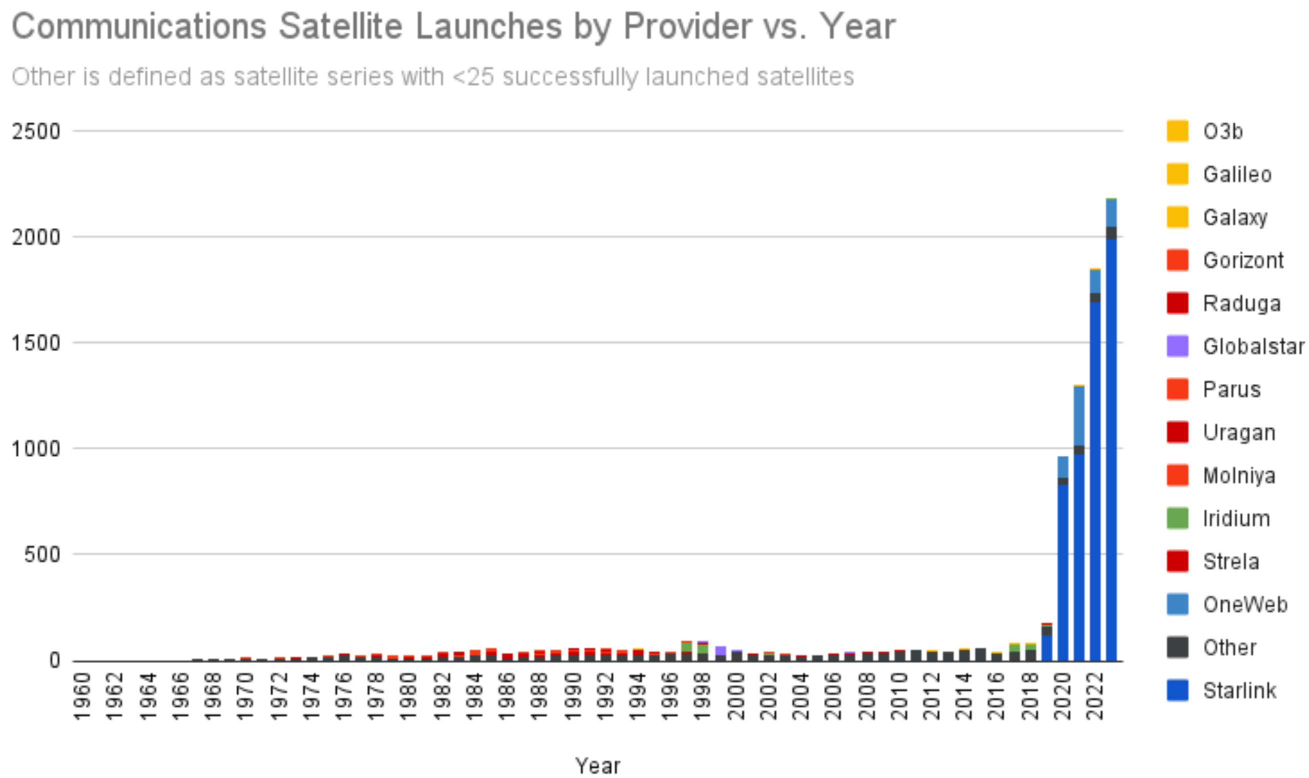
In my blog post from 6 days ago I wrote about the insights I gained along the way to chart out communication satellite bandwidth cost over time. As part of that blog post I created the chart you see above of communication satellites launched over time and described some of the the bumps you see on it. However, without any colour coding or categorization, it's difficult to parse. So, I suffering in Google Sheets for a few hours and categorized the data by Satellite Series so you could better visualize my points.

So, here's 4 fun charts that visualize the history of communications satellites with an explaination under each one of interesting points.

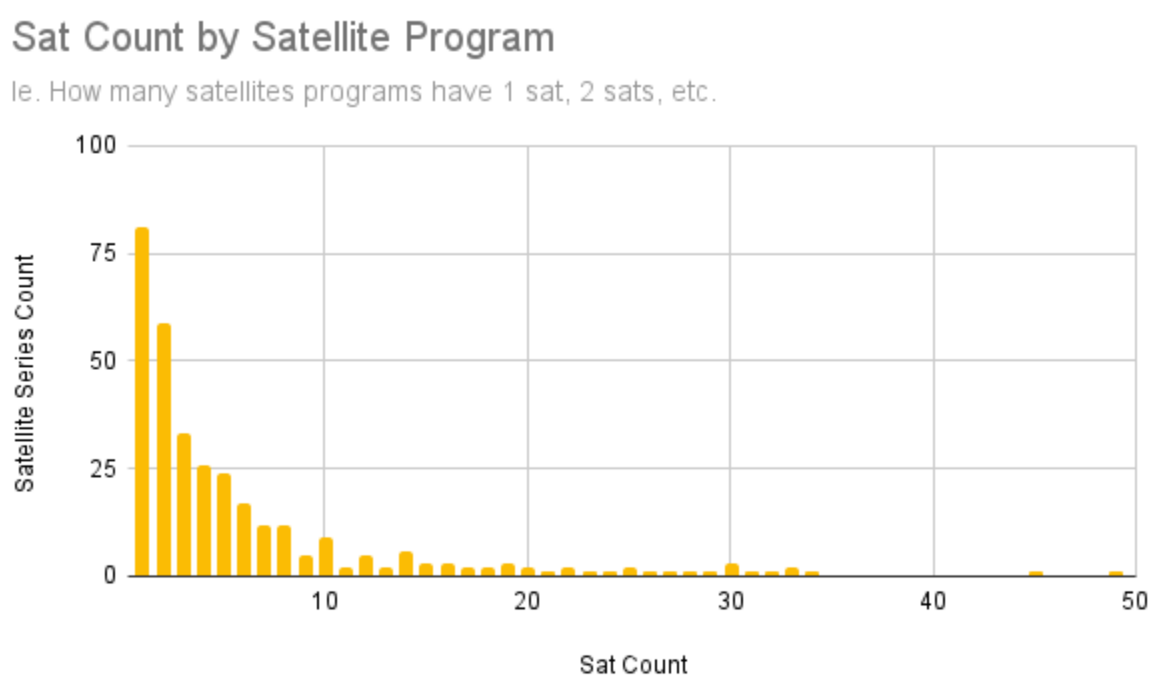


This is the same chart as the one above but with colour coding for each satellite series. Just appreciate for a minute how information dense it is now!

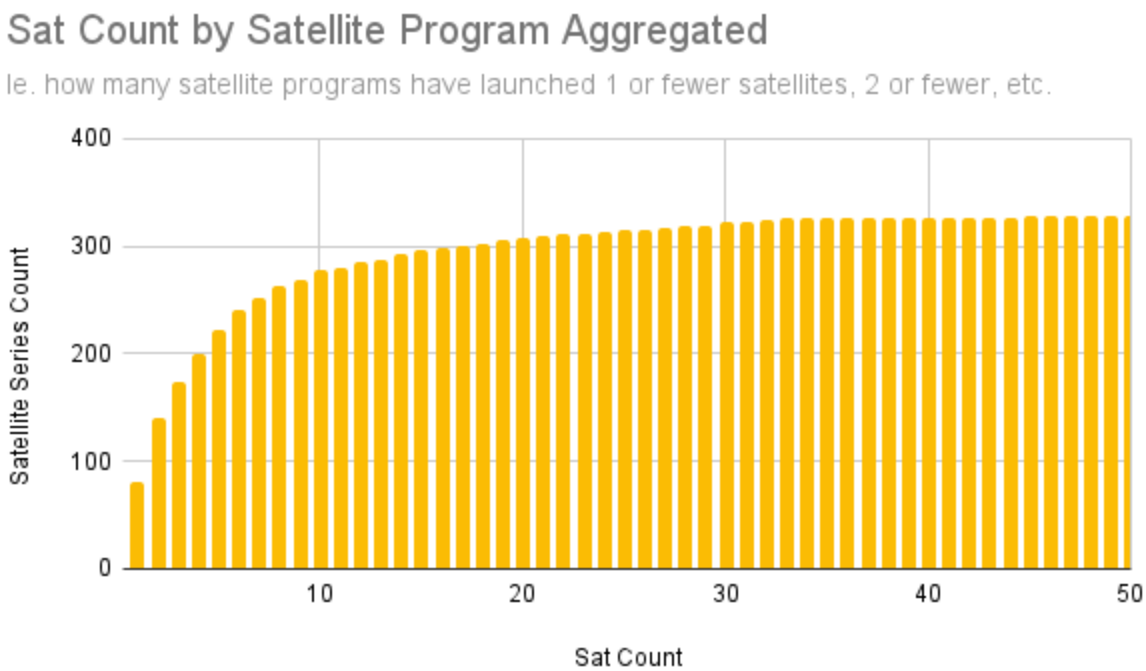
By removing Starlink, we can better see the history of Communications Satellite launches. You can clearly see the Iridium V1 bump from 1997-1998, and directly after it the Globalstar bump, which were Iridium's chief competition back in the day. This also illustrates the massive launch requirements for internet constellations vs. call constellations with the size of the Iridium and Globalstar bumps relative to OneWeb (and of course Starlink, but that's on a whole other level). All the red on the chart is all the other Soviet/Russia satellites and the yellow is western communications satellites and Galileo, which is the European GPS system.



This chart is the same as the one above but includes Starlink, notice how you can learn pretty much nothing except that yea Starlink is still winning and Elon is the most important living human.



Just for fun, you can see the power law distribution of satellite programs by the number of satellites launched. As you'd expect, most satellites are one-offs or part of a small series, you get what'd you'd expect after that, an exponential decay.



This chart is another way to illustrate the power law distribution of satellite programs as you can see how many satellite programs have launched 1 or fewer satellites, <=2, <=3, etc. The previous chart is the derivative of this function.

Subscribe

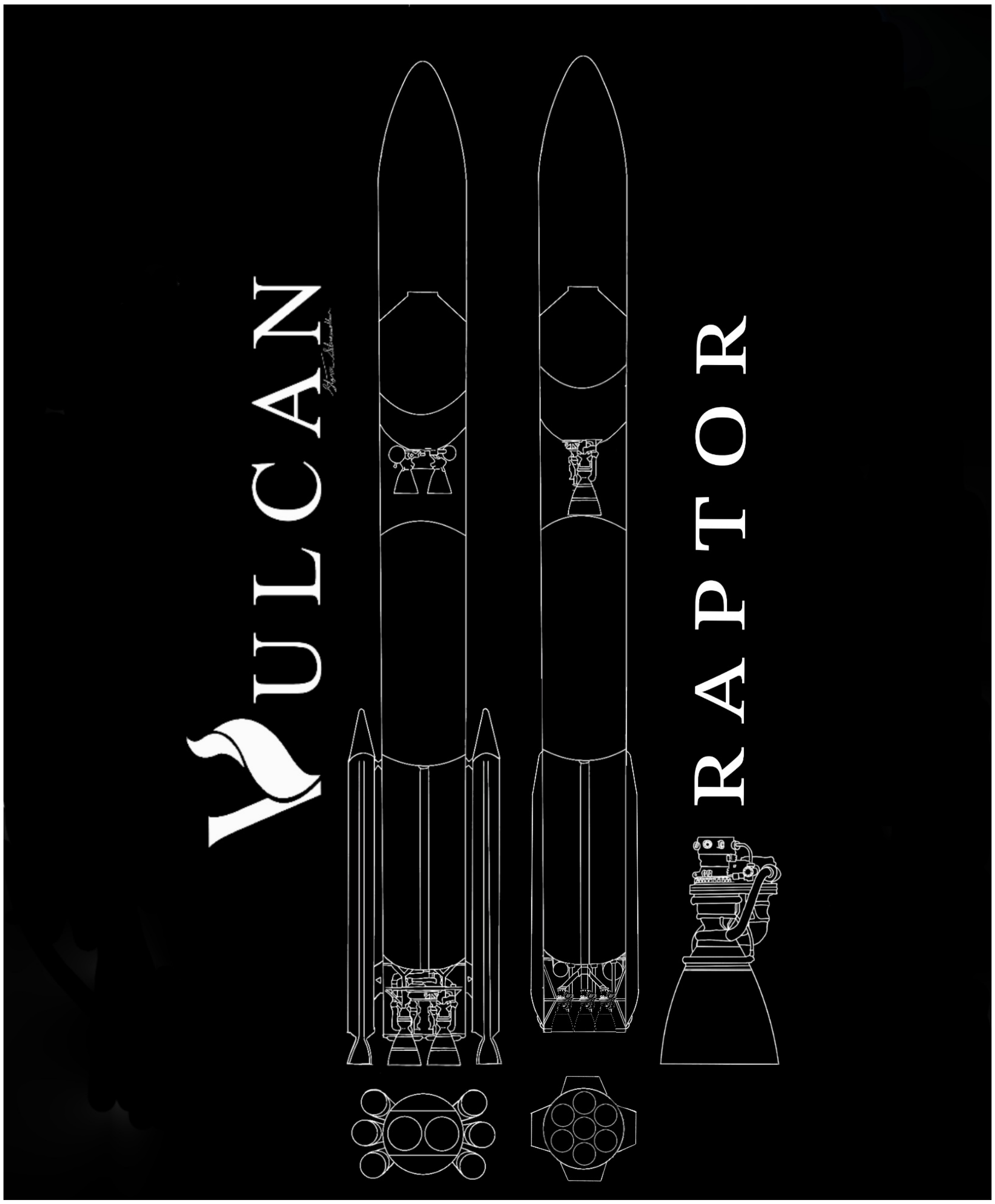
CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

A New Paradigm of Launch Vehicle Architecture Design Tradeoffs

Jan 12, 2025 • Christopher Kalitin



This image I got from @StormSilvawalk1's reply illustrates the two paradigms well. 10 separable vehicle parts (SRBs, stages, fairing etc.) vs. just 2 for Starship & Neutron.

We are currently living through a paradigm shift in launch vehicle architecture design. We are seeing old space companies like ULA or Ariane Space that are used to building highly customizable vehicles try to adapt to a new paradigm of reuse and high flight rates. Every day we continue to see these legacy perspectives bump up against new ideas that are optimized for different requirements. These companies once had extremely successful business models, but now they are struggling to adapt to the new paradigm of launch vehicle design, sometimes resulting in laughable concepts.

We're Seeing a Paradigm Shift From Customizable Vehicles to Vehicles Focused on LEO

The old paradigm of vehicle design was optimized for infrequent launches to a variety of orbits. The Evolved Expendable Launch Vehicle (EELV) program vehicles are great examples of this. The Delta IV and Atlas V were both optimized to have a wide range of configurations to meet the various target orbit and mass specifications that the US government laid out. This led to 11 flown configurations (according to Wikipedia) for the Atlas V and 5 flown configurations for the Delta IV. These configurations varied the number of boosters and size of the fairings and were the proper tradeoff for the requirements that Boeing and Lockheed Martin were given at the time.

The new paradigm ushered in by SpaceX's Falcon 9 is focused on reuse, high flight rates, and standardizing your vehicle for a specific orbit. Falcon 9 was originally optimized only to deliver the Dragon capsule to the ISS which is in Low Earth Orbit (LEO). Now, its primary use is launching the Starlink satellites to LEO. To a first approximation, Falcon 9's architecture is solely focused on getting as much mass to LEO as possible. Starship shows this goal even more clearly. It is a vehicle architecture that is completely focused on getting as much mass to LEO as possible. This means that through reuse and an extremely high flight rate they can achieve low cost, and then leverage refilling missions to get to Mars or other high-energy destinations in the solar system.

The Paradigms Decide Vehicle Architecture Optimizations

Both the old paradigm of vehicle customizability and the new paradigm of achieving low cost through reuse and high cadence are sound given their respective requirements (reaching EELV target orbits vs. getting tons of mass into LEO). However, issues start to arise when the companies that were dominant in the old paradigm try to adapt to the new paradigm.

Legacy space companies like ULA or Ariane Space are used to building customizable vehicles (varying booster counts or fairing sizes) because this was the most economical way to launch all the payloads they were tasked with launching. It is unfortunate to waste a full vehicle rated for 20t when your payload is only 10t, so instead you rate your base vehicle for 10t and add boosters onto it to be able to launch 20t. Atlas V took this to an extreme with some of its launches using only one booster (imagine that offset thrust)!

This approach has no place in the paradigm shift to reuse. Because the Falcon 9 is capable of booster reuse, SpaceX is not particularly impacted by whether their payload is 2t or 20t, they get the booster back either way so they have no reason to spend all the money required to create 10 different variants for different payload classes.

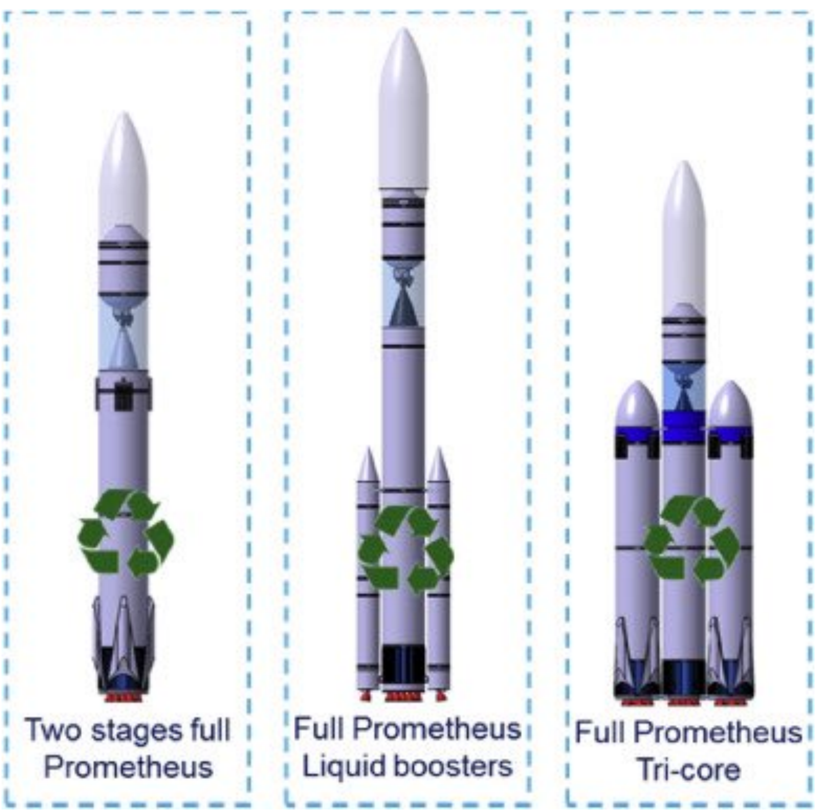
A defining characteristic of this paradigm shift is the approaches to lowering the cost of missions. The legacy model was to use as little hardware as possible for a given mission to save money (hence the 11 Atlas V configurations). The new model is to use a standard vehicle for all your missions to save money. This is because the primary cost savings of reuse come from high cadence (this was part of the reason the Shuttle was never cheap). Flying a standardized vehicle for all your missions has immense value in the cost savings the high cadence provides.

With the partially reusable Falcon 9, there are still some areas where SpaceX doesn't get cost savings from high cadence and they revert to the old model of using as little hardware as possible to decrease cost. For example, on Transporter missions, they use a shorter Merlin engine nozzle extension to save the cost of the Niobium.

Looking into the future, SpaceX is fully bought into reuse and the result is they are developing a single vehicle that they can throw at every problem. With Starship they're again going all in on a single vehicle and throwing it at every problem.

Everyone on X already understands all my points intrinsically through memes. A meme is worth a million words, and it spreads as a mind virus.

Not Having Full Buy-in To The New Paradigm Explains All The Foolishness We're Seeing



The Ariane NEXT concept illustrates why Ariane Space is not adapting to the new paradigm well.

There is this hilarious, infamous clip of an Ariane Space exec saying SpaceX's plans are a dream and if they ever come true that Ariane Space will be quick to follow. This is a phenomenon of believing the current paradigm of launch vehicle design will continue long into the future and a belief that Ariane Space would be fast enough to keep up with any paradigm shifts. Both of these points turned out to be false, hence why this clip is hilarious.

One of Ariane Space's concepts for a reusable vehicle is Ariane NEXT. The end goal of Ariane NEXT is a partially reusable 3-core vehicle, similar to Falcon Heavy. One of the interim steps described in a paper on the concept (Section 4.2) is to have smaller liquid boosters based on the same engine to increase the payload capability of the vehicle. There is also some discussion of a hydrolox upper stage, while the standard vehicle is methalox.

The Ariane NEXT concepts are an example of Ariane Space still thinking in the old paradigm of having a highly customizable vehicle. Instead of completely buying into having just a single vehicle (or at most 2 versions like Falcon 9 and Falcon Heavy), they want to have 3+ versions (single core, small boosters, large booster, plus a potential hydrolox upper stage). This is an approach that will not get them the cost savings of having a single standard vehicle (like the Falcon 9 booster).

The architecture of the Vulcan rocket shows that ULA is another old-space company that isn't adapting to the new paradigm well. Vulcan is a customizable vehicle in that it can have 0, 2, 4, or 6 solid boosters. Almost as an afterthought, they included some reuse in the design with their SAFER concept, that decouples, reenters (with an inflatable heat shield), and parachutes the engines back to Earth. This shows even less buy-in to the new paradigm, as they are seemingly trying to reuse as little of the vehicle as possible while keeping the ability to add solid boosters to increase payload capability.

The new paradigm of reuse incentivizes a single vehicle that can be reused as much as possible for all missions. ULA's Vulcan architecture is not optimized for this. Like I said in the first section, these design decisions make complete sense if you're operating in the old paradigm, but they are not the right tradeoffs for the new paradigm. Vulcan is optimized to launch US Military payloads through the NSSL program, but not to launch as much mass to LEO as possible which is the requirement for upcoming megaconstellations (See the Constellations - The Next Paradigm section). When launch is no longer extremely supply-constrained, it will become clear that Vulcan is not optimized for competing with vehicles optimized for economical reuse.

Conclusion

The fundamental issue with Ariane Space's and ULA's approaches is that the new paradigm of reuse and high flight rates incentivizes integrating as much onto a single reusable booster as possible. Don't have solid boosters, just make your core bigger and get good at reuse. Don't even have separable fairings! Integrate those onto a reusable stage so reusing them is even easier!

ULA and Ariane Space have clearly not committed to this new approach and are hanging on to the old paradigm of having a highly customizable vehicle. This approach may work for them in the short term as they have government support, but in the long term, it will become clearer and clearer that this approach will not be commercially competitive with other vehicles optimized for reuse.

Updates

[Update Jan 12 2025]
Casey Handmer explains the issues with increasing the number of vehicles your architecture uses in this blog post. It's a good read to get more context on this issue and described the issues with increasing the complexity of you architecture well.

Another way to consider the paradigm shift I described is architecture complexity vs. launch cost (ie. do you make more vehicle configurations or just make a single vehicle that is oversized for most tasks that'll be cheaper through reuse).

Subscribe

CKalitin
x.com/CKalitin

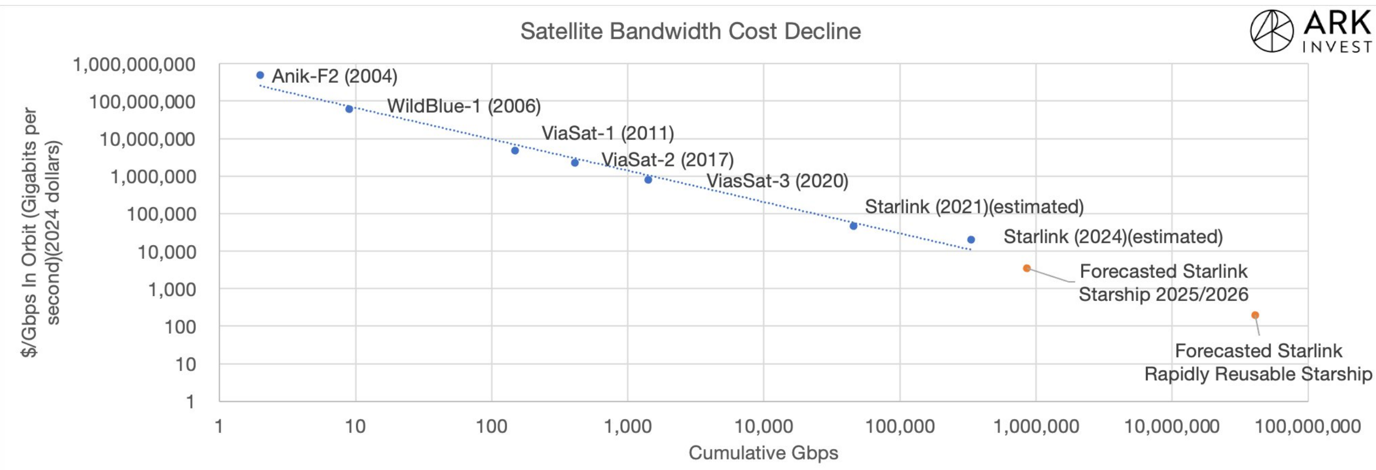
Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Estimating Cost per Mbps of Communication Satellites Over Time

Jan 9, 2025 • Christopher Kalitin

Find my raw data here.

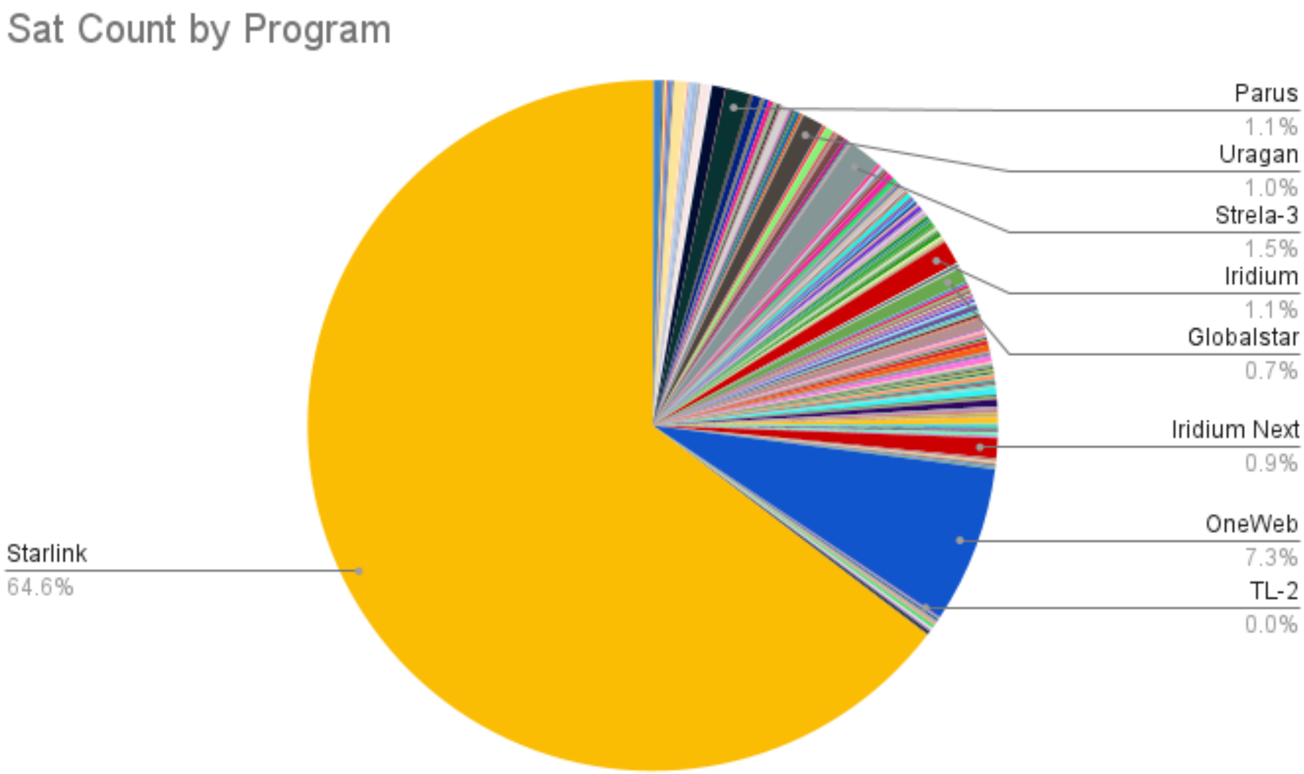


This is the Ark Invest chart that inspired this blog post.

Three hours ago I saw (Space) Case Taylor reply to a post Sam Korus of Ark Invest made in which he showed a chart of satellite bandwidth cost over time. You can see this chart above and it clearly shows the exponential cost decline of satellite bandwidth over time, the beauty of learning rate in production. As an exercise and an opportunity to learn more about the communication satellite industry, I decided to derive a similar chart myself.

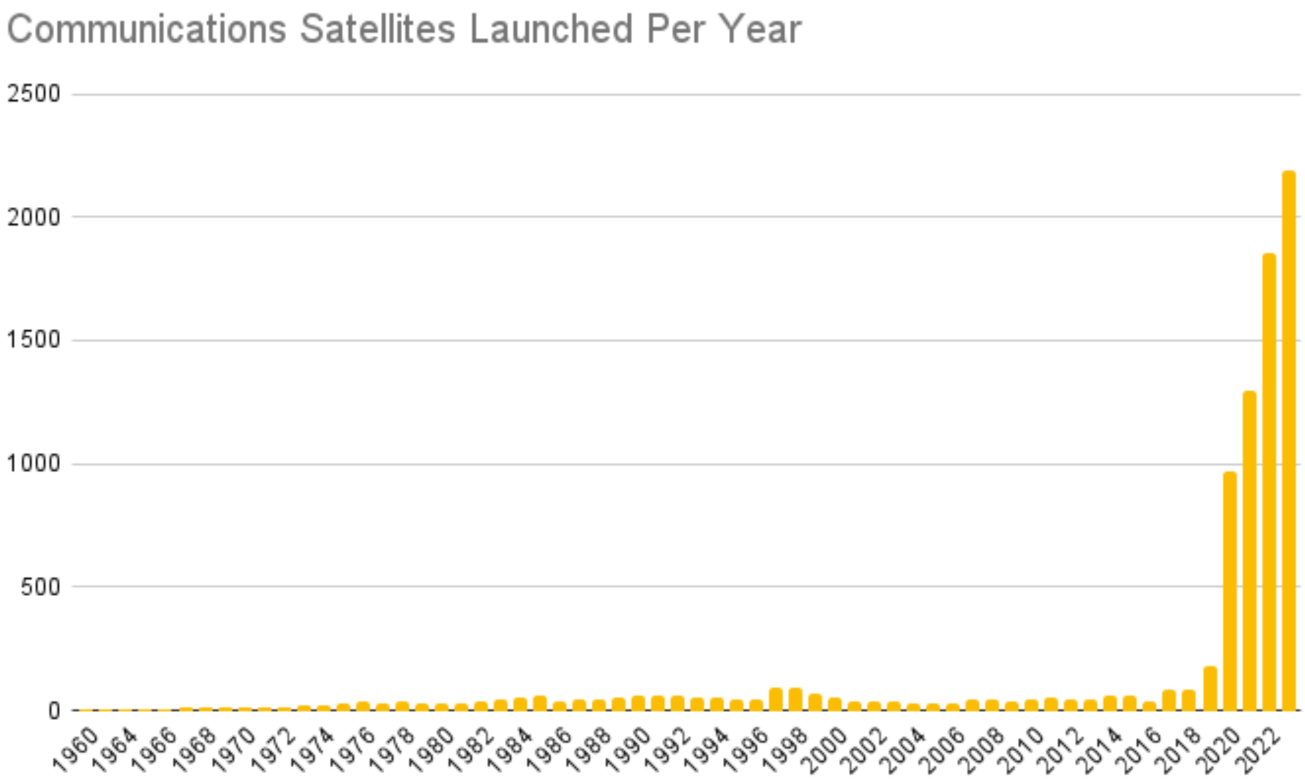
Read the sequel to this blog post to get better charts the describe the history of Communications Satellites.

Using Jonathan Mcdowell's Dataset



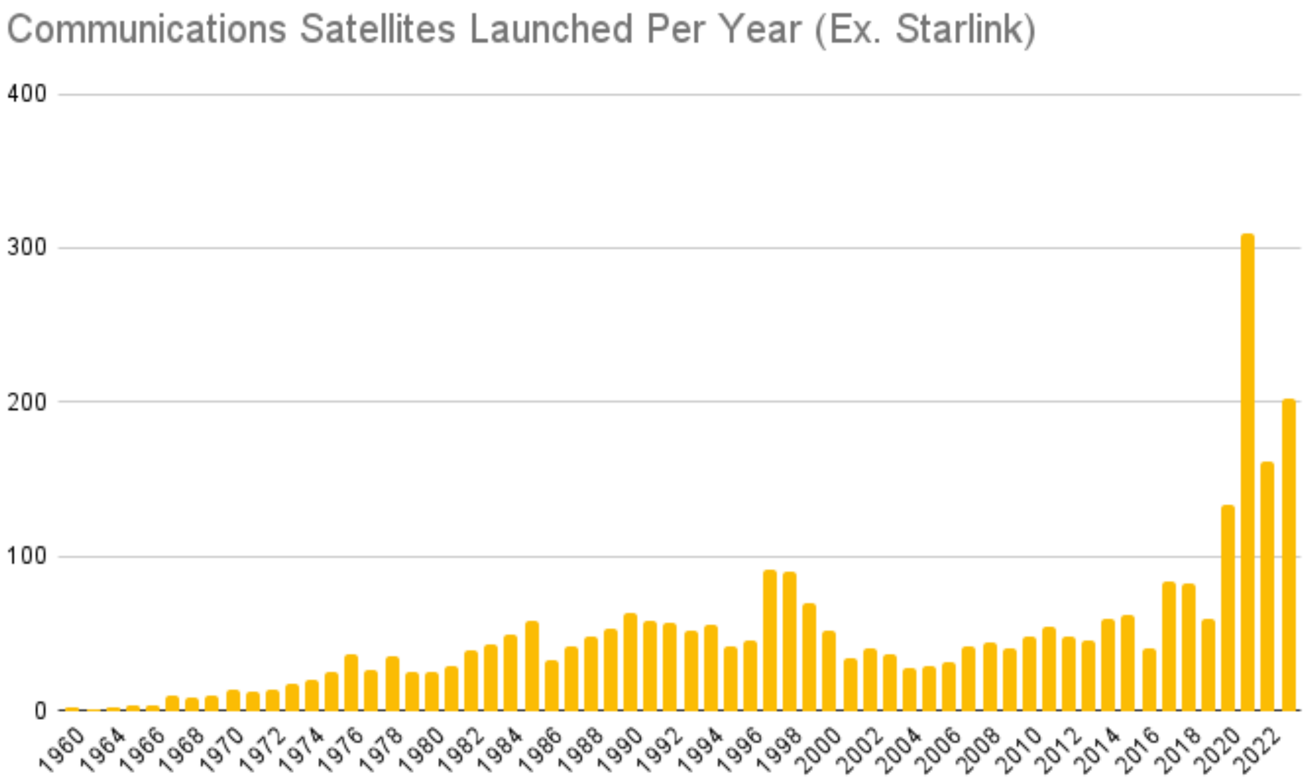
This shows share of total communications satellites launched in all of history by satellite program.

The first step was to filter Jonathan Mcdowell's dataset to include only communication satellites. I've worked with his dataset for over a year and already did this a while ago. This resulted in a list of 8,691 communications satellites successfully launched between 1960 and 2023 (inclusive), the first of these being Transit 1B on April 13, 1960. This data isn't directly pertinent to the result I'd like to get, but the journey is important and there are insights to be gained from this, these insights are the purpose of this section.



That massive wall post 2020 is Starlink.

As is a common theme in trying to analyze the satellite industry, SpaceX's massive market share makes it very difficult to extract any insights from the raw charts, and if you want to learn anything other than "yea SpaceX is still winning" you have to exclude them from the dataset.



Without Starlink, you can get a feel for the history of communications satellites.

Above you can see communications satellites launched per year excluding Starlink and there are a lot of interesting stories behind all the bumps and increases.

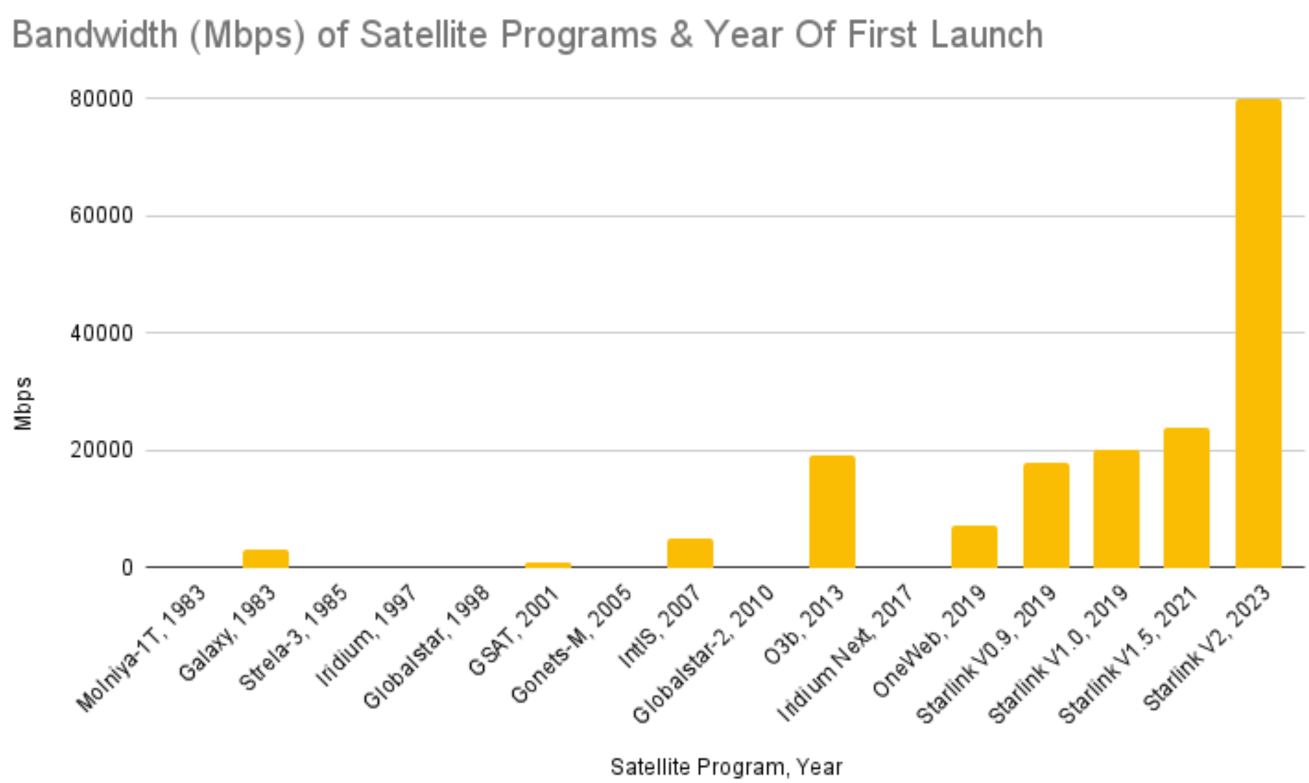
The 1997-1999 bump is mainly due to the initial Iridium satellites. The majority of these satellites were launched by the Delta II out of SLC-2A, Iridium practically owned all launches out of this pad for a couple years. They had 40% of all launch contracts in the world during this time. I've been reading Eccentric Orbits and there are a lot of fun stories about Iridium's early days. Another one is that in the lead up to the initial launches, the GPS IIR-1 launch failed and the Delta II was grounded for a few months, you've probably seen the video of this and Scott Manley has a great video on it, as it's one of the most spectacular launch failures in history. It's also the origin of the "we have had an anomaly" meme circulating on X.

The smaller 2017-2018 bump is the Iridium NEXT launches on Falcon 9. Post 2020, the large increase is due to all the OneWeb launches on Soyuz, Falcon, and ISRO's LVM3.

Getting Price & Bandwidth

With the data on every communications satellite ever successfully launched, I narrowed the dataset down to only programs with more than 20 satellites launched of a single series. This narrowed it down to 33 series of satellites. Some of these could be combined (eg. Jonathan Mcdowell lists many subcategories of Starlink satellites when I only needed V0.9, V1, V1.5, and V2 mini) or ignored (it's difficult to get pricing data on Soviet Molniya satellites launched in the 70s). I was left with a set of 16 satellite programs/series to analyze. With the help of Grok, ChatGPT, and the internet I got price and bandwidth estimates for each of the satellites including manufacturing and launch. Many of the early satellites didn't have good data and the LLMs hallucinated often, but the prices generally seemed to be in the right order of magnitude, which is sufficient for this analysis. Later satellites like Iridium, OneWeb, and Starlink had good data available, so the important data points would be reasonably accurate.

Here I'll note that this is not the approach Ark Invest took, they used a set of GEO & Starlink satellites with good data on them and fit a trendline to the price per Mbps. For my next attempt this is probably what I'll do, start with satellites that have good price data instead of satellites with many launches.



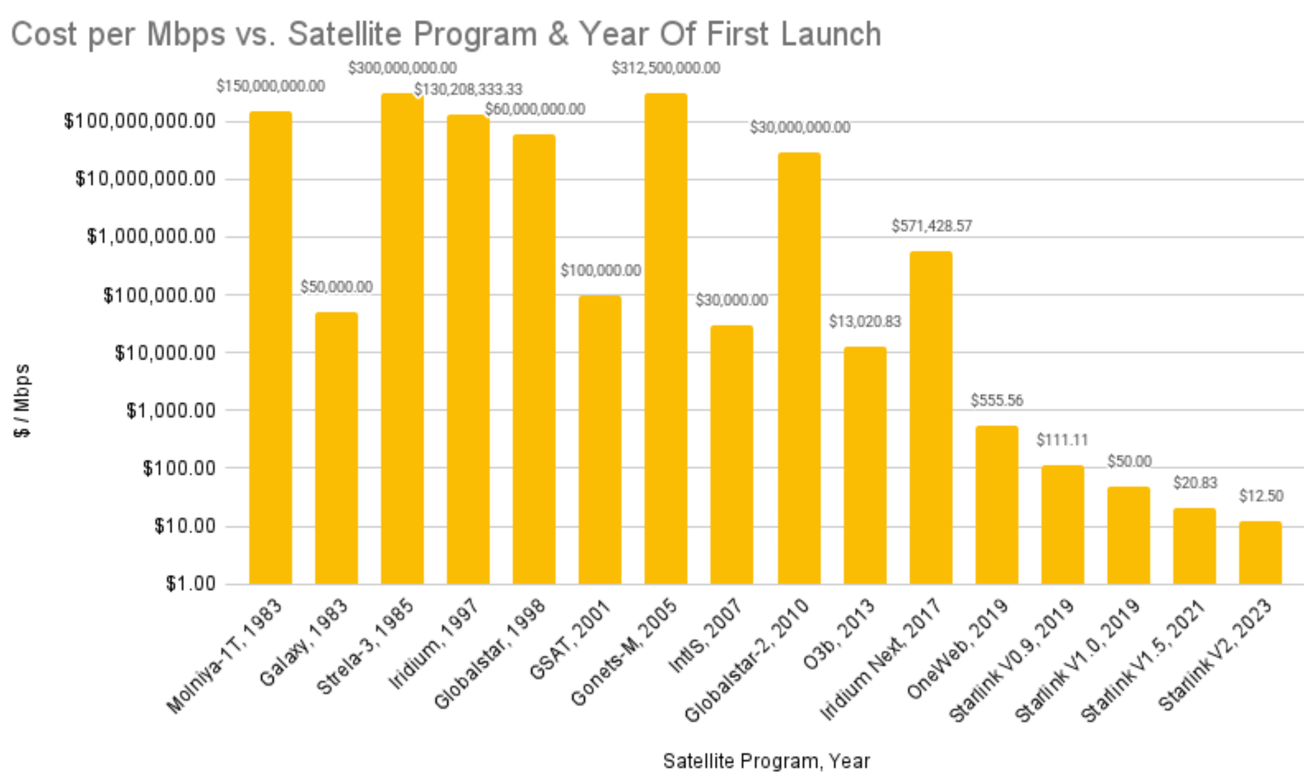
You can see that bandwidth per satellite has been increasing exponentially over time.

Above you can see the bandwidth of single satellites from various programs over time. There are four primary categories here, early communications satellites (Molniya), early LEO satellite phone constellations (Iridium, Globalstar), GEO satellites (O3b, Intelsat (IntlS on the chart), SES Galaxy), and LEO internet constellations (OneWeb, Starlink).

The early satellites like Molniya and Iridium supported such low data rates that they're not even visible with Starlink on the chart. Each Iridium call was on the order of a few kilobits per second for audio transmission, and the Molniya satellites were even worse. In contrast, internet requires megabits per second, a minimum of 6 orders of magnitude more data.

The GEO satellites also had high total throughput compared to the early LEO constellations. These satellites provided TV, data, and phone service to entire continents, while the early LEO satellites served a few thousand customers at a time. This meant total data throughput to customers could be much higher for single satellites, especially given that many people could tune into a single TV channel at once, meaning a single signal from a satellite could be received by far more people than would be involved in a phone call. This is part of the reason I didn't include any GPS/GNSS satellites in this analysis, it's not clear what bandwidth means in the context of a GPS satellite where you're just broadcasting a constant signal to be received by a huge number of receivers.

Results



We see an exponential decline in cost per Mbps over time on this logarithmic chart.

In the previous section, I described the 4 major categories of satellites that I included in the dataset. Because these satellites have different purposes, we see ~3-4 orders of magnitude difference in the 1983-2017 set of satellites. This is because the GEO satellites can deliver far more bandwidth to customers than the LEO satellites like Iridium, so the cost per Mbps differs a lot. This is similar to the issues you encounter when using $/kg as a metric in launch vehicles, Electron is several times higher than F9 but still has customers because its customers have different needs.

Again, this is a reason to do the analysis in the way Ark Invest did instead of the way I did. Maybe Sam Korus of Ark also ran into similar issues as I did in doing his research and he course corrected before publishing a graph. I'm not profesional enough to not show my work, so this blog post is full of interesting intermediary results.

Post Iridium Next / OneWeb in 2017 and 2019, we see an exponential decline in the cost of bandwidth with LEO broadband satellites. Remember that these numbers are mainly illustrative of the shape of the graph since SpaceX doesn't give us precise internal launch cost and satellite manufacturing numbers, but the trend is clear. The cost of bandwidth from space has been exponentially declining and is likely to continue as satellite technology continues to improve, the scale of the satellites increases with Starlin V3, and competition enters the arena.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Using SCPI and PyVISA to Automatically Collect Data for ADC Characterization

Dec 29, 2024 • Christopher Kalitin

This blog post is a copy pasted Monday Update I wrote for UBC Solar. So here are some defitions/explanations of terms & names:

People:

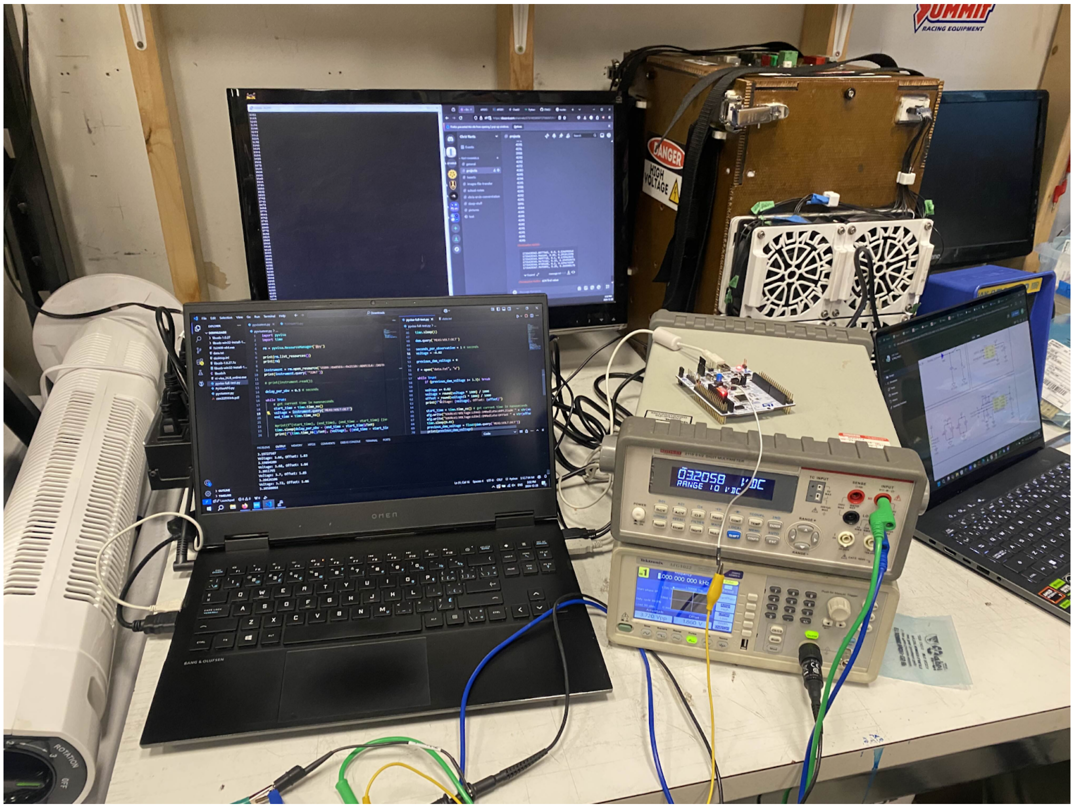
1. Saman - UBC Solar Electrical Lead
2. Mischa - Previous UBC Solar Electrical Lead and Previous BMS Lead
3. Krish - Current UBC Solar BMS Lead

Terms:

1. Monday - Our project management software
2. SCPI - Standard Commands for Programmable Instruments
3. AFG - Arbitrary Function Generator
4. DMM - Digital Multimeter
5. Nucleo - STM32 Nucleo Micro Controller
6. HPPC - Hybrid Pulse Power Characterization (Battery test)

Context: This was part of a UBC Solar project to characterize the current sensor in our battery pack.

Github Project Link containing SCPI Python scripts & STM32 Project: here
Raw Data and Chart: here



My final testing setup

This morning Saman suggested I find a way to use our Digital Multimeter (DMM) and other tools to automatically collect data for ADC characterization. In my previous Monday update I described how we can feed the output of the DAC into the ADC pin to automatically characterize an ADC over a range of voltage values. This plan falls through because I found that the DAC can't be trusted to output a specific voltage value and needs to be characterized itself. The solution is to use external trustworthy equipment to complete the automatic characterization.

I started by figuring out how to program the DMM to automatically record and print values. SCPI (Standard Commands for Programmable Instruments) is the standard communication protocol to do this and there is a python library called PyVisa that allows for simple scripting. This took hours to set up and was most of my day, but it's simple if you know what you're doing, which now I do. If we plan to do similar tests in the future it would be useful for me to write a wiki page on how to set it up (automatic data collection like this was suggested for HPPC testing).

My goal was to automatically generate a voltage and read it with both our DMM and a Nucleo. This way we would assume the DMM is trustworthy and could characterize the error in the Nucleo's ADC.

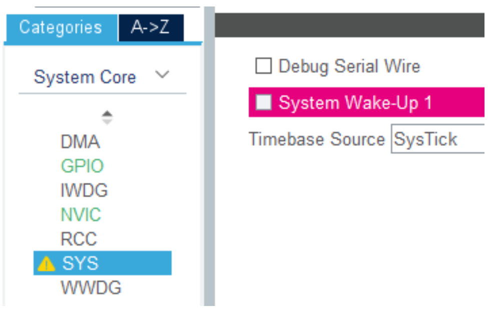
Our Power Supply doesn't support SCPI and can't be programmed, but our Arbitrary Function Generator (AFG) does support SCPI.

I wrote a script that made the AFG output a constant voltage that was read by the DMM. I ran this in a loop from 0 to 3.3V with a 0.02V step size. This 0.02V step size was use for the final of 5 tests

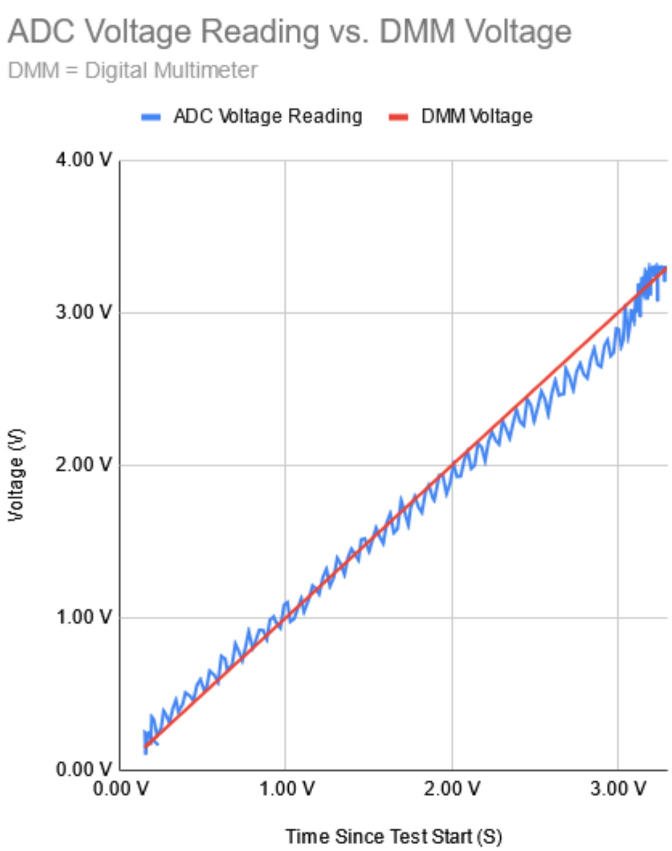
I did with this script and is what generated all the charts in this update.

Note: the AFG doesn't have a settings for a constant voltage, so I made it a pulse with a 99.9% duty cycle which appears to be close enough to a constant voltage for our purposes.

A step is done once per second, so a full test took ~4-5 minutes. This long step length is used so that the Nucleo, AFG, and DMM will be roughly synchronized without any communication between them. To illustrate, It's easier to click start on both in a 1 second window than a 0.01 second window.

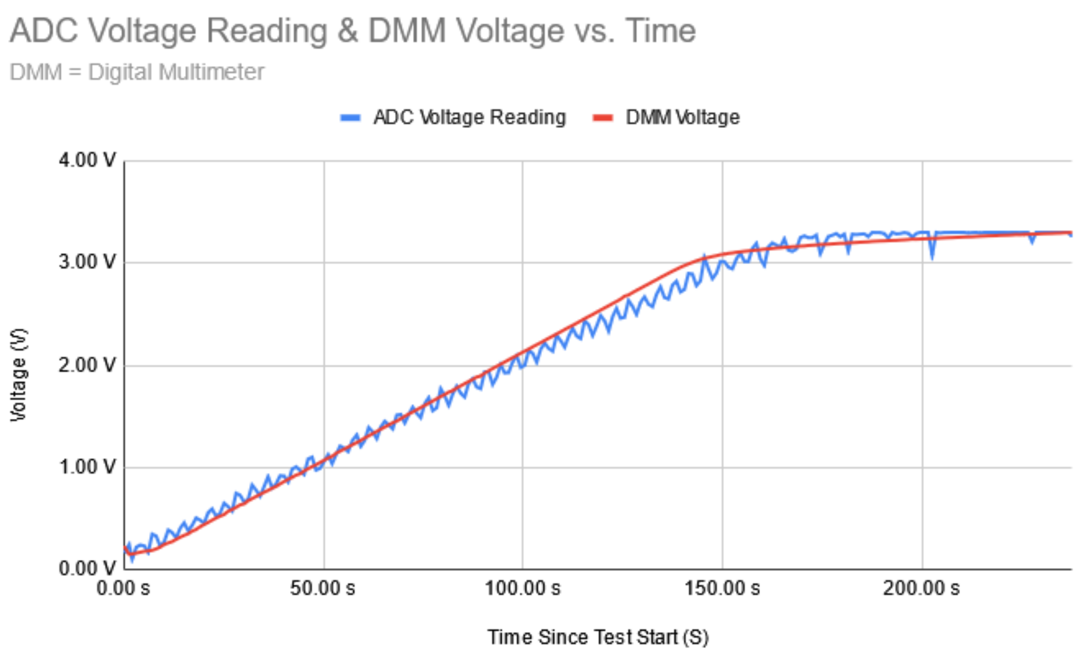


Side note: When Krish and I were testing the ECU a few weeks ago we lost the ability to upload code to the STM32 chip, we struggled for 1.5 weeks and Mischa solved our problem in 5 minutes by showing us the Debug Serial Wire toggle in STM32 Cube IDE. This came in handy when I was testing with the Nucleo today, without it my test would've been stopped in its tracks. Thx Mischa.



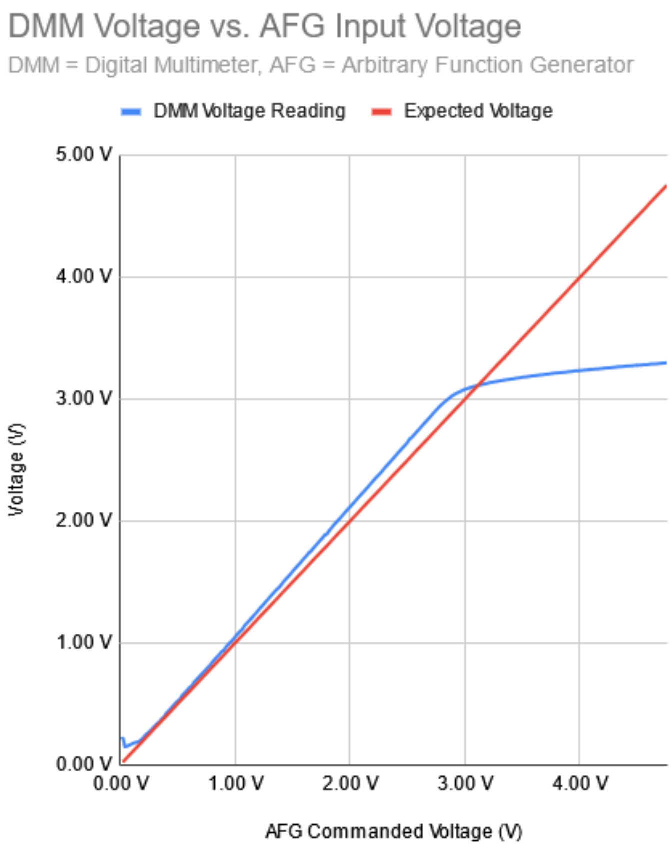
This is the graph that can be used to characterize the Nucleo's ADC

As you can see in the graph above, there is a strange zig zag pattern in the graph of the ADC Voltage Reading. This isn’t exactly noise because it is very regular. There’s probably a fundamental reason for this that I am not aware of, but in future testing averaging can be used to get a smoother result. Also, I didn’t remember to measure the voltage reference for the Nucleo’s ADC so I assumed it was 3.3V.



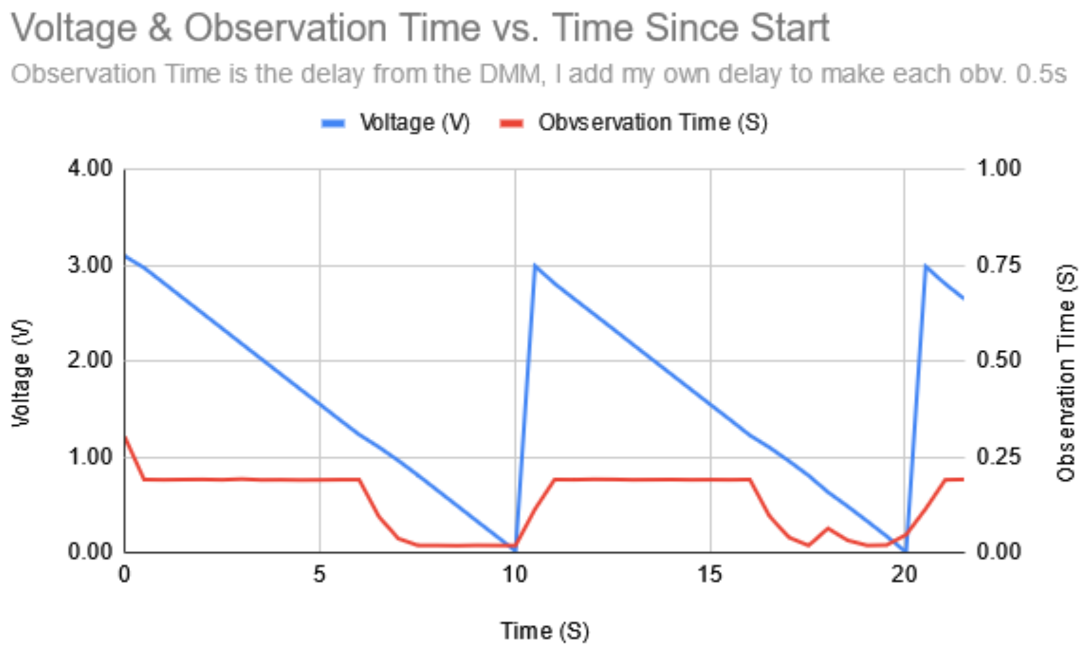
This graph shows observations versus time, not observation vs. observation like the previous graph. You can see recording over the 4 minute test.

The graph above shows that the rate of voltage increase after 3V is lower than that below 3 volts. My script increases voltage by 0.02V every second, so the voltage I am commanding the AFG to output does not match the voltage it is outputting.

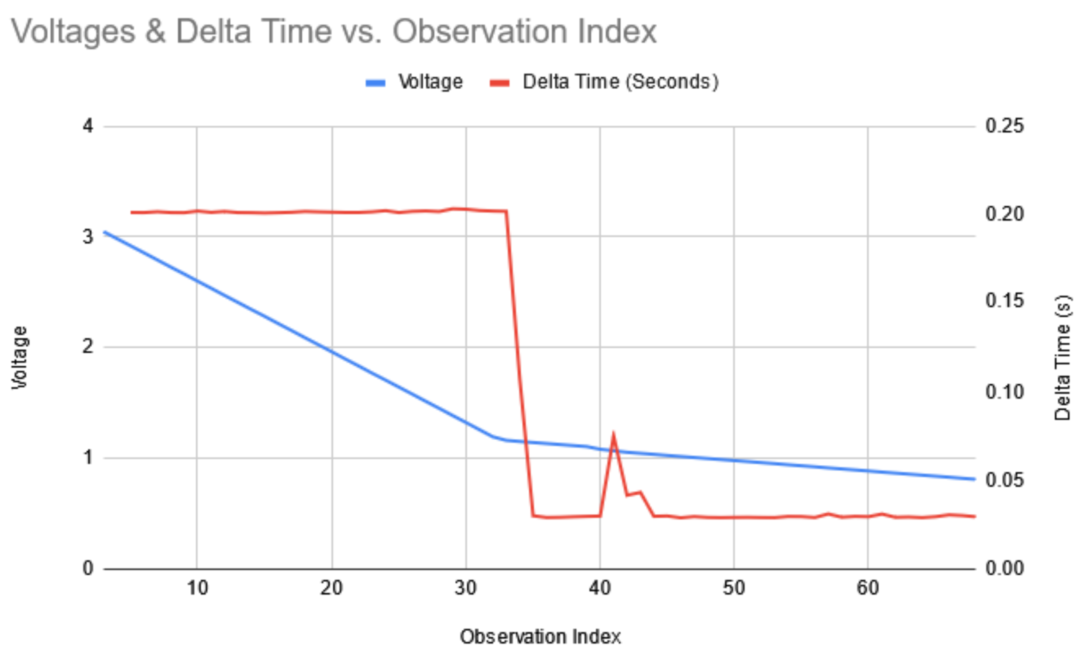


</i>Above ~3V, the AFG stops outputting the voltage it is commanded to.</i>

When plotting the AFG output voltage versus the expected AFG output voltage, we see what we expected to, after ~3V the AFG stops outputting the voltage is is commanded to. With some manual testing I found that when I set the AFG to 5V manually it does in fact output ~5V (~5.12V really, the AFG isn't perfectly precise). So, there must be an issue with programming the AFG or with my code that causes the discrepancy between expected and real results.



Notice that at above 1.23V the observations take ~10x longer than below 0.81V.



When we plot against observation index instead of the time stamp of the observation, we see the increase rate of observations after ~1.2V.

During initial testing I noticed that the DMM outputted values far faster lower voltages than at higher voltages. Because I assumed a constant time delta between observations, this discrepancy showed up as a bend in the graph of voltage vs. observation index which you can see in the second chart above (Voltages & Delta Time vs. Time Stamps).

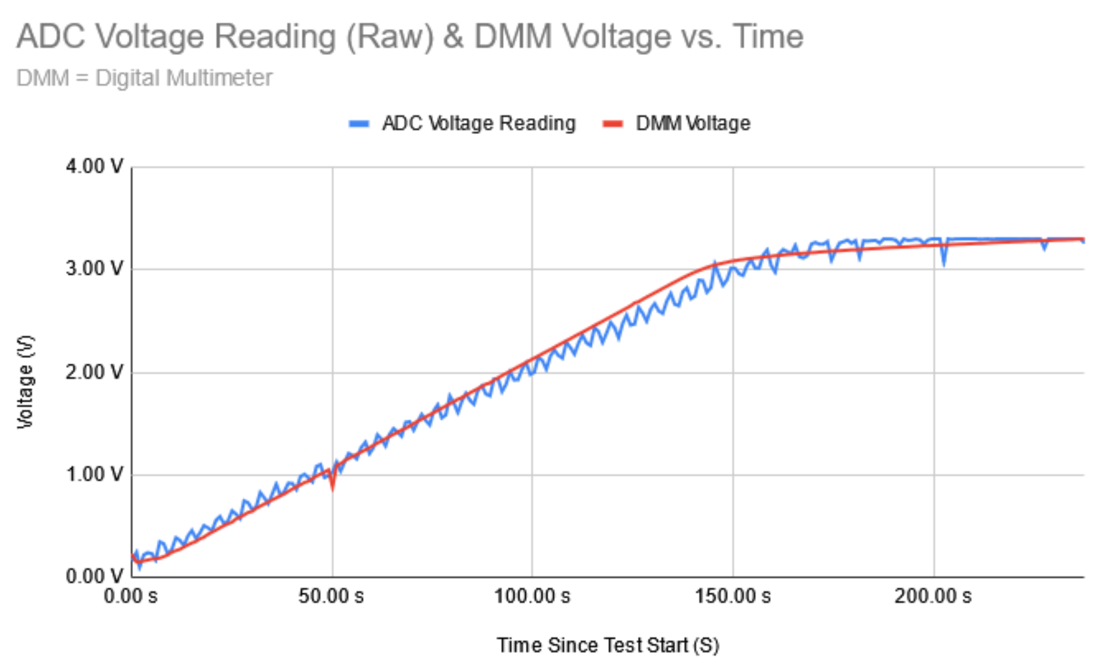
I solve this problem by requiring each observation to be done at a constant cadence. I require 1 second per observation. To meet this requirement I take an observation and record how long it look, by using the expression *delayTime = 1 - obvTime* I find the remaining delay required to space each observation out by 1 second. The longest observation sets the minimum time required for all other observation if we want a constant cadence, so our theoretical minimum is ~0.3s. Read the Python script here.

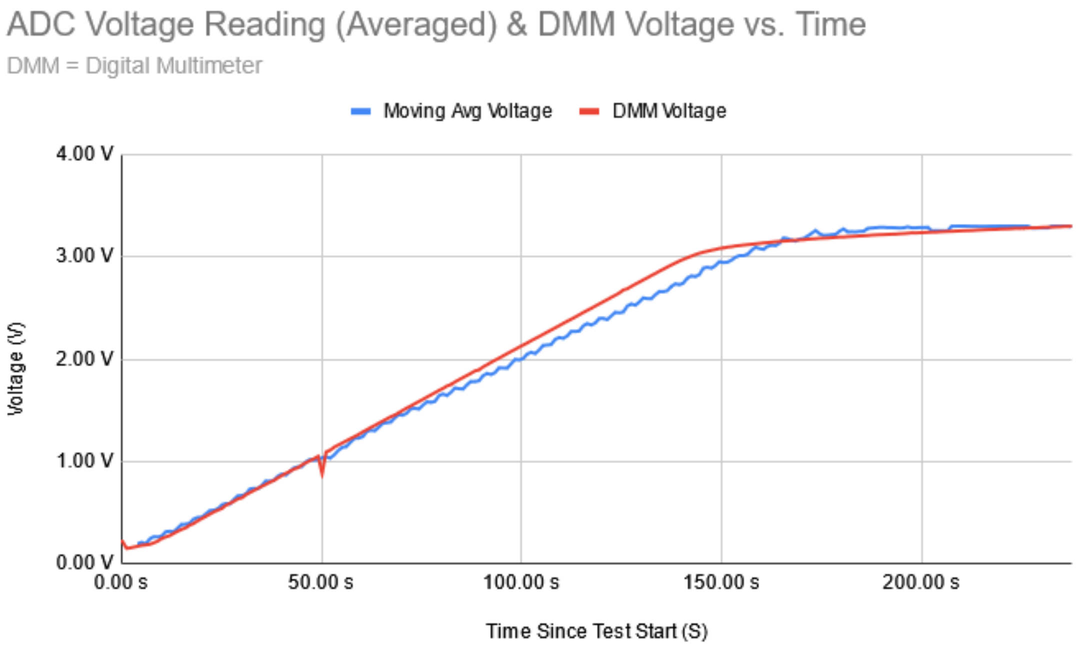
Overall, this test made me learn a lot about SCPI and interfacing with electronic instruments. For our goal of more accurate current data, an automatic test setup like this isn't strictly required. Recording values manually may take ~30 minutes while it took me ~5 hours today to set up automatic data recording - this could be reduced to ~1 hour of programming/setup for future tests. Although this might not be on the critical path for a more accurate current sensor, learning SCPI has value in and of itself and this knowledge may prove valuable in the future.

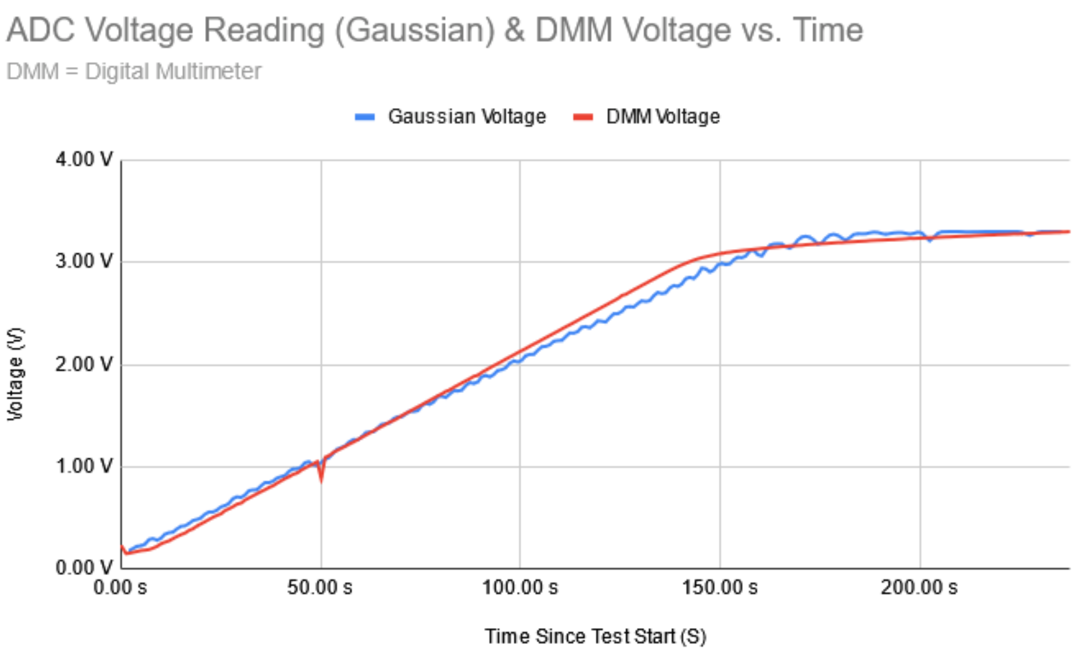
Updates:

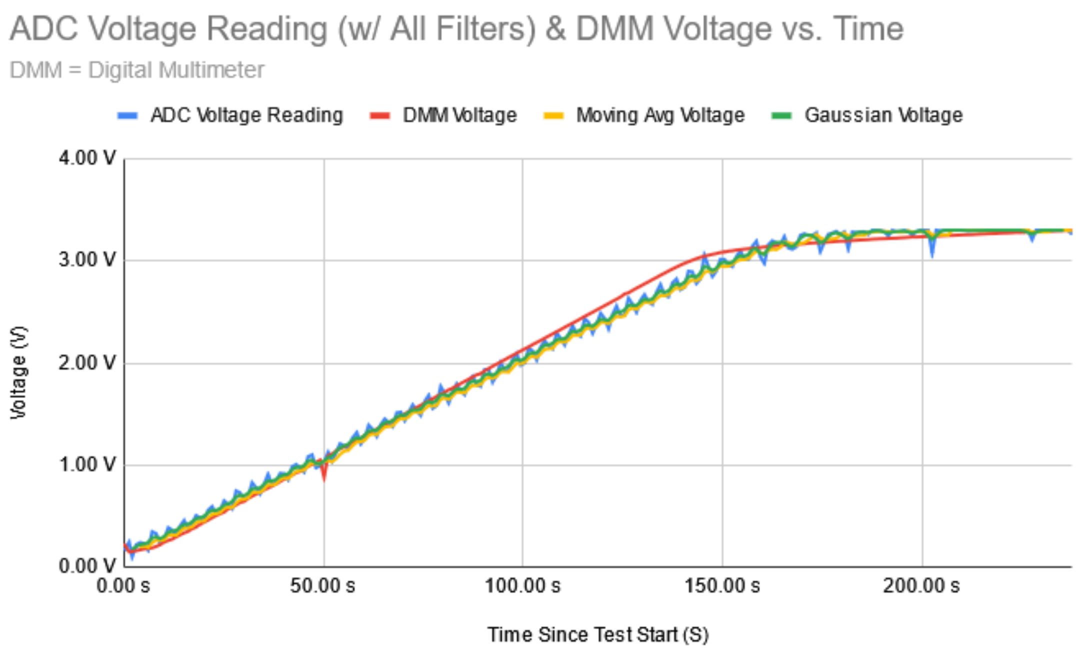
At Saman's request, here are ADC Voltage Reading & DMM Voltage vs. Time graphs but with averaging and gaussian filters applied.

The average takes the sum of the previous 4 values and the current value and divides by 5. The Gaussian uses a 0.06, 0.24, 0.4, 0.24, 0.06 filter centered on the current value.




ADC Voltage Reading (Averaged) & DMM Voltage vs. Time
DMM = Digital Multimeter
Moving Avg Voltage
DMM Voltage
4.00 V
3.00 V
2.00 V
1.00 V
0.00 V
Voltage (V)
0.00 s
50.00 s
100.00 s
150.00 s
200.00 s
Time Since Test Start (S)


ADC Voltage Reading (Gaussian) & DMM Voltage vs. Time
DMM = Digital Multimeter
Gaussian Voltage
DMM Voltage
4.00 V
3.00 V
2.00 V
1.00 V
0.00 V
Voltage (V)
0.00 s
50.00 s
100.00 s
150.00 s
200.00 s
Time Since Test Start (S)



Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

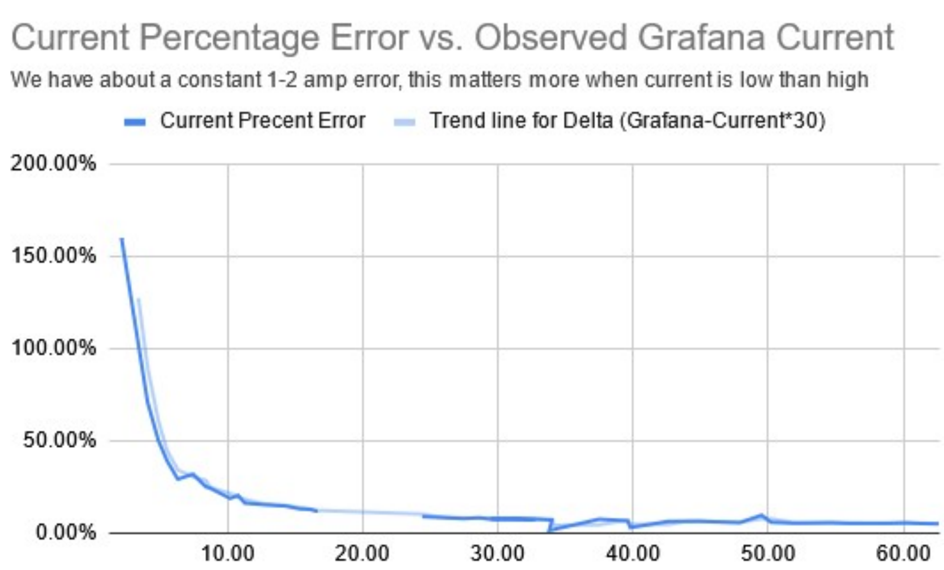
Characterizing the ESP32's ADC

Dec 27, 2024 • Christopher Kalitin

Introduction

I joined the UBC Solar design team in September as a member of the Battery Management System team. I get to talk to wizards every meeting and work on very exciting projects, which makes this is the best team I've ever worked with and most exciting project I've ever worked on. Completely amazing projects and people.

For the previous few months, we've been trying to diagnose an issue we've been having with the current sensor on our battery pack. We use the HASS-100S hall effect sensor and it was giving us inaccurate current data during competition in the summer. We spent months diving deep into what could cause this and the project took many turns. We started with characterizing the current sensor itself, then realized the ADC pin was the issue and tried characterizing it, then realized all ADC pins were off which is where we arrive today. This process took months mostly because we didn't understand these systems from first principles, now that I've done this testing, fixing this kind of issue in the future will take 10x-100x less time.

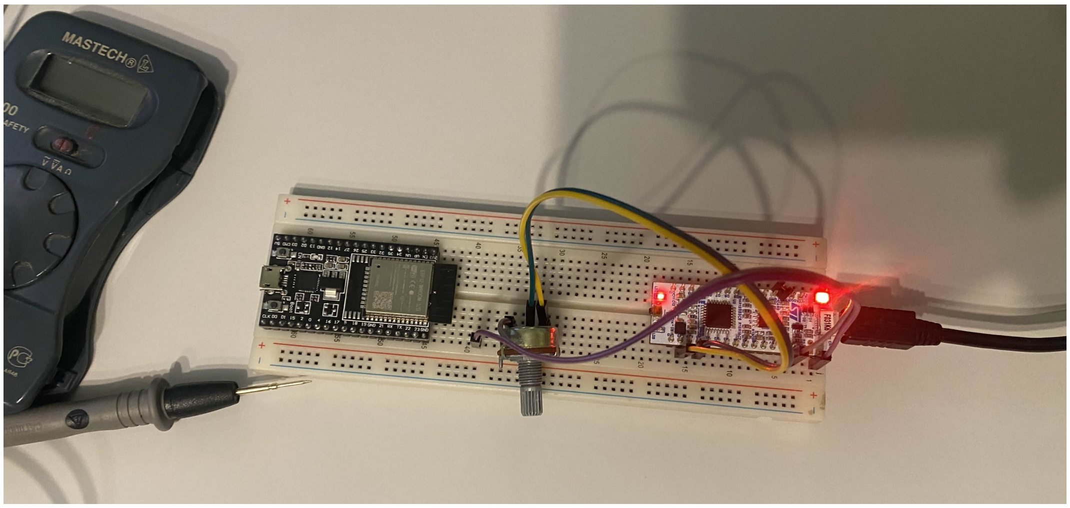


Error Graph from our initial characterization of the current sensor

A few days ago I had my last final exam for the term, so the real work of figuring out how microcontrollers actually work could begin. After bouncing around a few of my existing projects I decided to characterize the ADCs on my STM32 F031K6T6 and ESP32-WROOM-32D to get a better understanding of how to use them and learn more about how they work along the way.

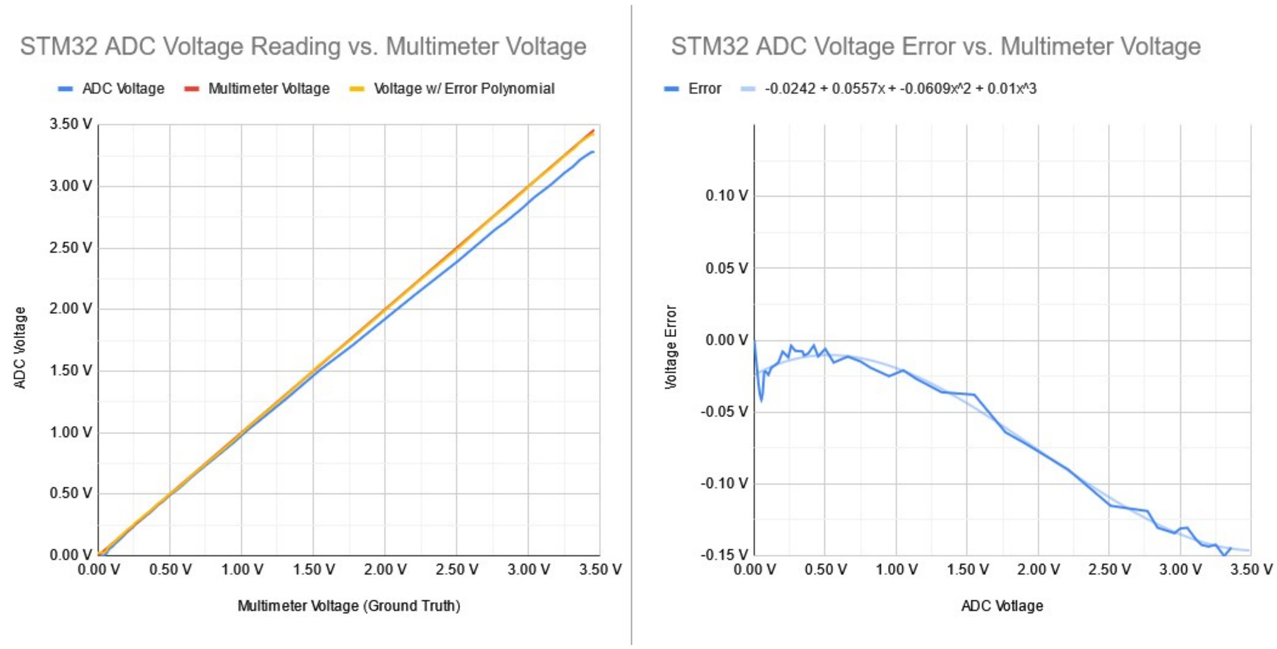
All test data can be found here.
My code can be found here, if you look through previous commits you'll find earlier test code I wrote for this project.

Initial Testing

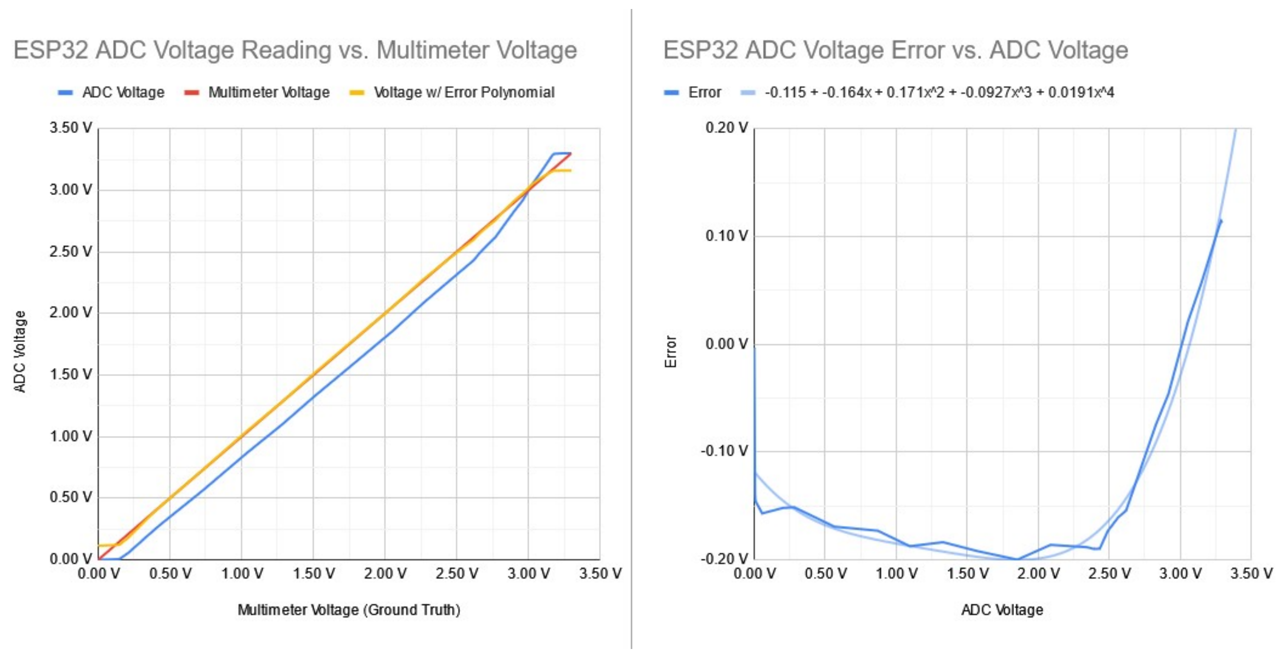


My Initial Testing Setup

I initially wanted a graph of ADC voltage vs. True voltage (as measured by a multimeter) for both microcontrollers. Both have 12-bit SAR ADCs so this testing is applicable to the work on the UBC Solar battery pack. I used a potentiometer to vary voltage between 0-3.3 to get a full range of values. I have no power supply (yet) so I must work with the tools I have. It doesn't look like the potentiometer introduced much error when comparing these results to those later in testing, it only introduced some noise.



Blue Line = ADC Observed Voltage, Red Line = True Voltage (what we expect), Yellow Line = ADC Voltage plus Error Polynomial

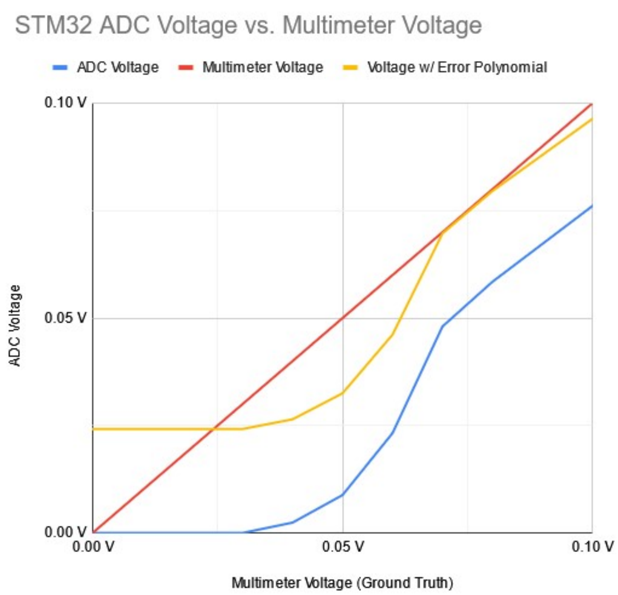


Blue Line = ADC Observed Voltage, Red Line = True Voltage (what we expect), Yellow Line = ADC Voltage plus Error Polynomial

The shape of the ESP32's ADC voltage graph perplexed me because it appears the first value it observed was 0.14V, and it stopped at 3.18V. Some googling & talking to Grok showed that below 0.1V or above 3.1V the ESP32's ADCs are not accurate. In the UBC Solar battery pack, we give the current sensor a 1.8V reference voltage to offset the readings. So, we are right in the middle of the accurate range of the ESP32's microcontroller. STM32 chips have a similar range in which they are meant to operate accurately.

With the ADC observed voltage and the expected voltage, I fit a polynomial to the error using

Google Sheet's built-in function and plotted the voltage + error polynomial on the graph. Note that I didn't redo the test here with the error polynomial applied in firmware, I just added it to the spreadsheet values. With the error polynomial applied, the STM32 and ESP32 ADCs became accurate to within 0.01V in most of the range (0.5V - 3.0V). The reason this range doesn't exactly match the expected 0.1V - 3.1V range that the ESP32 chip should be accurate in is because of improper application of the error polynomial, I should have deleted values that are outside the range I want.



Zoomed in view of the STM32 ADC Voltage vs. True Voltage

This zoomed-in view of the same STM32 ADC Voltage vs. Multimeter Voltage graph from above shows a strange non-linearity point in the graph. Above a voltage of 0.07V, we get a mostly linear error. However, below this, we see this strange non-linear exponential-looking part of the graph. I don't understand Electrical Engineering or Physics at a low enough level to describe this, but it tells us that the reasonable lower range of the STM32's ADC is 0.07V.

1.2V AAA Battery Test		
Multimeter Volta	ADC Reading	ADC Voltage
1.20 V	1270	1.02 V
1.20 V	1290	1.04 V
1.20 V	1310	1.06 V

Testing with a 1.2 V Battery

I also tested with a 1.2V AAA battery to ensure that the potentiometer was not an issue and got results that were similarly inaccurate. These results also had a lot of noise. Note that I converted the ADC Reading into a voltage in the spreadsheet with the formula Voltage = (ADC Reading / 4095) * 3.3. This assumes the reference voltage is 3.3V and converts the ADC reading assuming the conversions: 0 = 0V and 4095 = 3.3V.

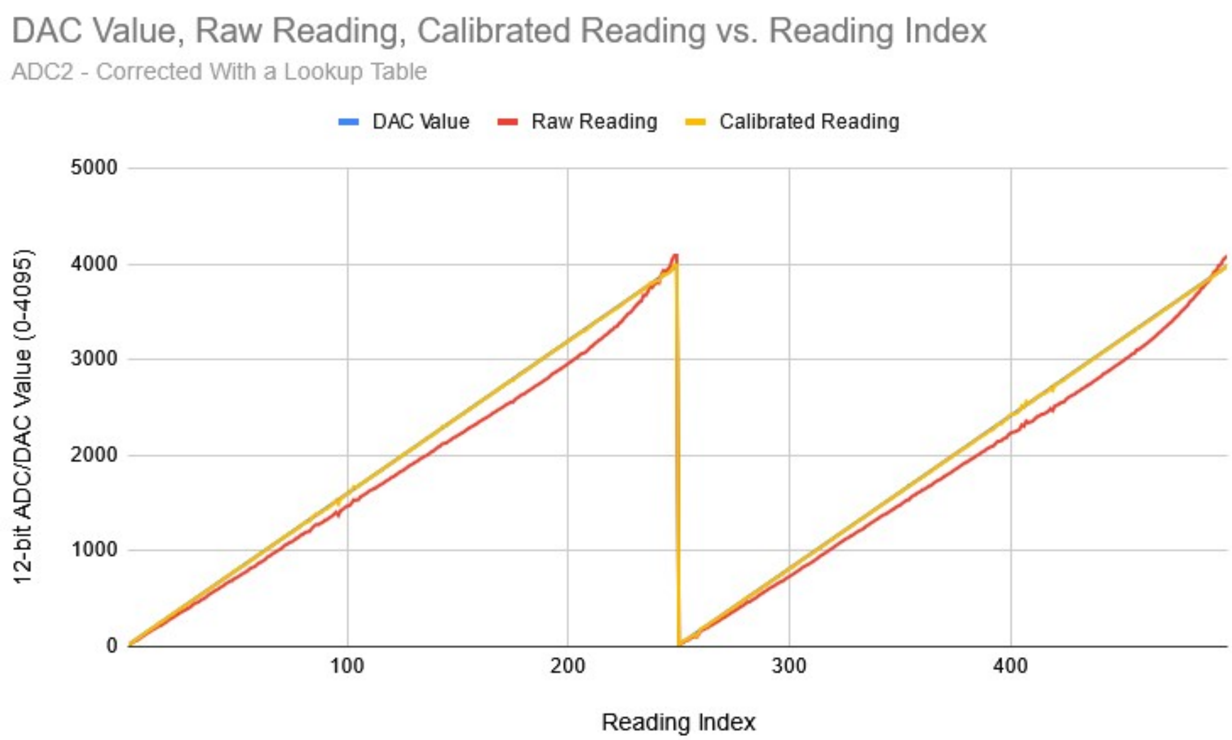
Using the esp32-adc-calibration Github repo

```
3885.0000,3885.8000,3886.6001,3887.0000,3887.8000,3888.6001,3889.0000,3889.8000,3890.2000,3890.8000,3891.3999,3892.0000,3892.8000,3893.0000,3893.8000,
3894.3999,3895.0000,3895.8000,3896.0000,3896.8000,3897.3999,3898.0000,3898.8000,3899.0000,3899.8000,3900.3999,3901.0000,3901.6001,3902.0000,3902.8000,
3903.2000,3904.0000,3904.6001,3905.0000,3905.8000,3906.0000,3906.8000,3907.2000,3908.0000,3908.6001,3909.0000,3909.8000,3910.0000,3910.8000,3911.2000,
3912.0000,3912.6001,3913.0000,3913.8000,3914.0000,3914.8000,3915.2000,3915.8000,3916.6001,3917.0000,3917.8000,3918.0000,3918.8000,3919.2000,3919.8000,
3920.6001,3921.0000,3921.6001,3922.0000,3922.8000,3923.0000,3923.8000,3924.2000,3924.8000,3925.3999,3926.0000,3926.6001,3927.0000,3927.8000,3928.0000,
3928.8000,3929.0000,3929.8000,3930.2000,3930.8000,3931.3999,3932.0000,3932.6001,3933.0000,3933.8000,3934.0000,3934.8000,3935.0000,3935.8000,3936.2000,
3937.0000,3937.8000,3938.2000,3939.0000,3939.8000,3940.0000,3940.8000,3941.6001,3942.0000,3942.8000,3943.6001,3944.0000,3944.8000,3945.3999,3946.0000,
3946.8000,3947.3999,3948.0000,3948.8000,3949.3999,3950.0000,3950.8000,3951.2000,3952.0000,3952.8000,3953.2000,3954.0000,3954.8000,3955.2000,3956.0000,
3956.8000,3957.2000,3958.0000,3958.8000,3959.2000,3960.0000,3960.8000,3961.2000,3962.0000,3962.8000,3963.3999,3964.0000,3964.8000,3965.3999,3966.0000,
3966.8000,3967.3999,3968.0000,3968.8000,3969.2000,3969.8000,3970.6001,3971.0000,3971.8000,3972.2000,3973.0000,3973.6001,3974.0000,3974.8000,3975.3999,
3976.0000,3976.8000,3977.0000,3977.8000,3978.3999,3979.0000,3979.8000,3980.0000,3980.8000,3981.3999,3982.0000,3982.8000,3983.2000,3983.8000,3984.6001,
3985.0000,3986.0000,3986.8000,3987.3999,3988.0000,3988.8000,3989.6001,3990.0000,3991.0000,3991.8000,3992.3999,3993.0000,3993.8000,3994.6001,3995.0000,
3996.0000,3996.8000,3997.3999,3998.0000,3998.8000,3999.6001,4000.0000,4002.6001,4004.8000,4007.0000,4009.3999,4011.8000,4014.0000,4016.2000,4026.80
};

void setup() {
    dac_output_enable(DAC_CHANNEL_1);        // Enable DAC on pin 25
    dac_output_voltage(DAC_CHANNEL_1, 0);    // Setup output voltage to 0
    analogReadResolution(12);
    Serial.begin(500000);
    while (!Serial) {}
}

void loop() {
    for (int i=1; i<250; i++) {
      dac_output_voltage(DAC_CHANNEL_1, i);    // DAC output (8-bit resolution)
      delayMicroseconds(100);
      int rawReading = analogRead(ADC_PIN);                          // read value from ADC
      int calibratedReading = (int)ADC_LUT[rawReading];    // get the calibrated value from LUT

      // Run Serial Plotter to see the results
      Serial.print(F("DAC = "));
      Serial.print(i*16);
      Serial.print(F(" rawReading = "));
      Serial.print(rawReading);
      Serial.print(F(" calibratedReading = "));
      Serial.println(calibratedReading);
      delay(10);
    }

}
```

This is the code from the Github repo I used, notice the line count and the massive array at the top that stores the lookup table.

In researching I found this Github repo that uses the ESP32's DAC (Digital to Analogue Voltage Converter) to get the chip to characterize itself. If you have a known voltage output from the DAC, you can pipe this directly into the ADC and find your error in firmware instead of on a spreadsheet. This Github repo then takes the error values and feeds them into an Arduino .ino program (ESP32's can run using the Arduino IDE) to get a lookup table of errors. Essentially, it records 256 values and for each range in between them, it calculates the error. With these ranges, you can use them as a lookup table where you input your ADC value as a key and get the expected ADC value back for that particular ADC voltage.

This lookup table approach required 16 KB of flash memory on the ESP32. If we really wanted to do this on the UBC Solar battery pack we could shrink the lookup table to 128 values and probably have enough flash left over for the other programs. However, this is still a very stupid idea. We aren't software devs that can throw memory and compute at all of our problems.



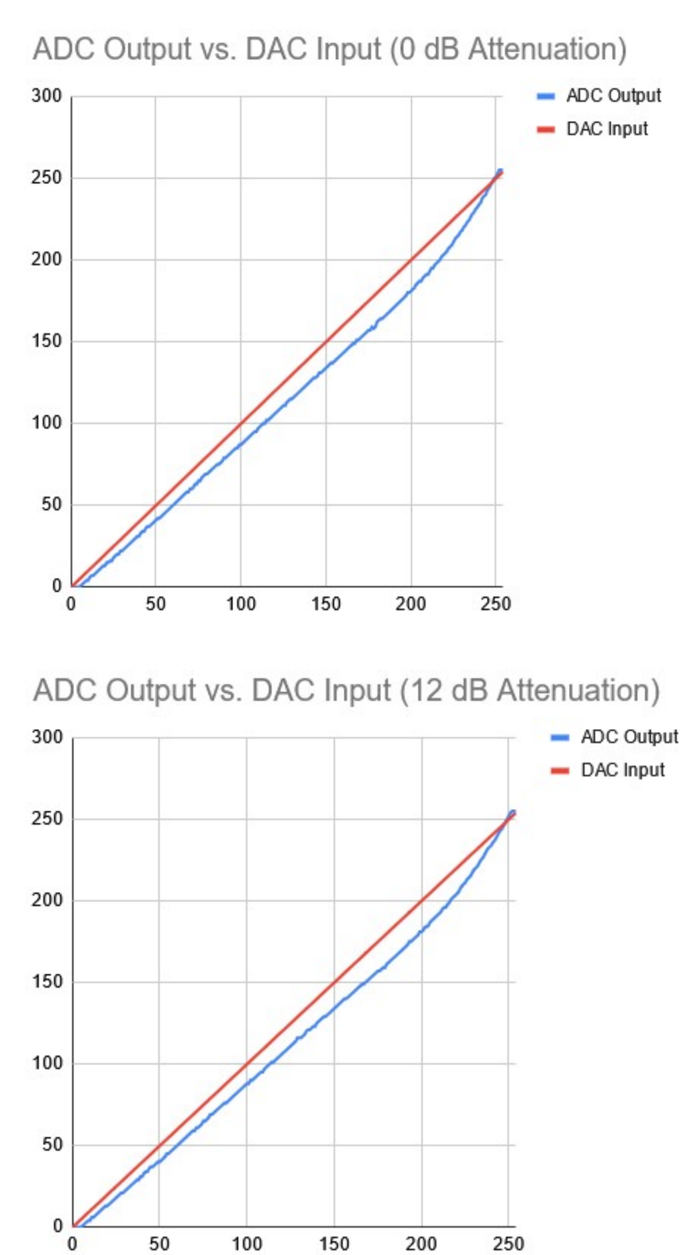
Blue Line = DAC Output Voltage, Red Line = Unadjusted ADC Reading, Yellow Line = ADC Reading Adjusted with Lookup Table

The shape of the error looks very similar in this test and the calibrated reading is spot on with what we expect. However, I later found out that the output of the DAC cannot be trusted. When testing with a multimeter, its output was inaccurate in a similar way to the ADC. So, this kind of test where you use the DAC to calibrate the ADC is not possible on my ESP32 chip.

I initially wanted to write my own self-calibration script for my STM32, but after realizing the DAC may not be trustworthy and the fact that unlike UBC Solar's STM32 chip mine doesn't have a DAC, I decided to abandon this idea. If on Solar we find that the DAC on our chip is trustworthy, this could be a good way to calibrate the ADCs on any of our boards, not just the ECU (Elithion Control Unit) in the Battery Pack.

Testing Attenuation

In the README for the esp32-adc-calibrate repo, the author mentions that reading the ADC Calibration documentation is a good first step before implementing the code. I read it and found the esp_adc_cal_characterize() function that allows you to set the attenuation, bit width, and other parameters for either of the two ADCs on the ESP32-WROOM-32D. There's no automatic ADC calibration, but I could set parameters and record the results.

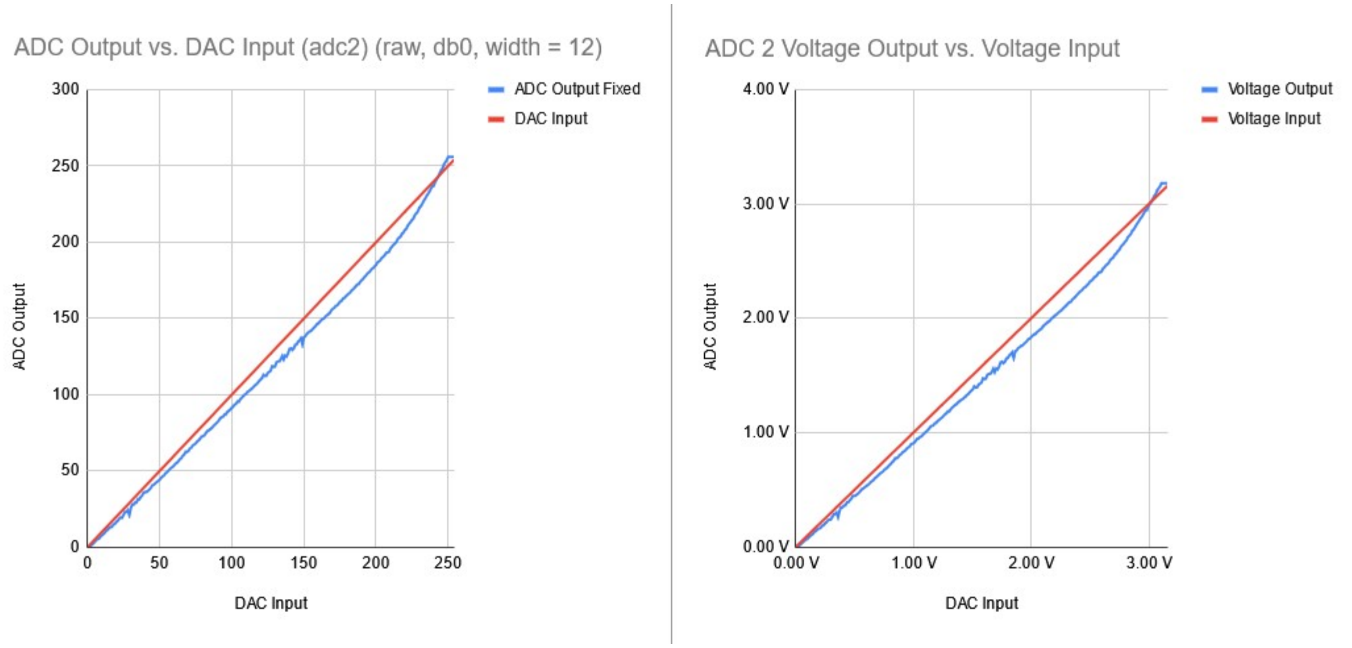


Little difference was found between all the attenuation values.

I wrote some firmware inspired by the Github repo above that automatically tested the ADC given a range of DAC values. This was before I found out the DAC is not trustworthy to output specific voltages within ~0.1V in most of the range while I needed ~0.01V precision.

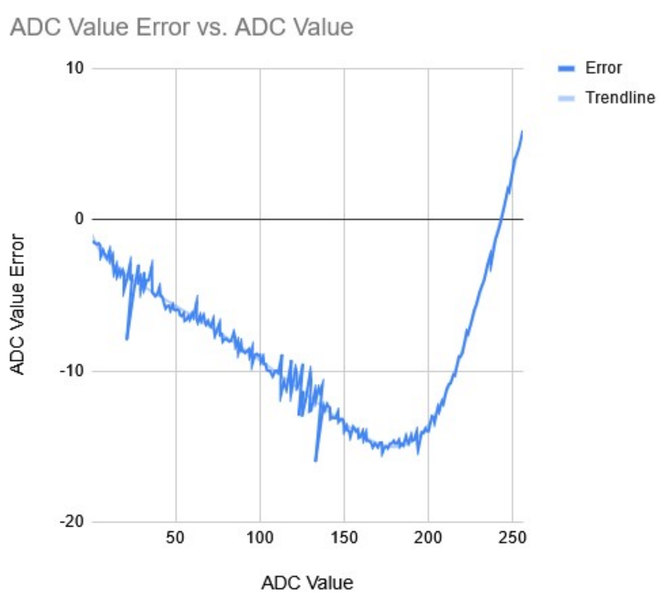
With that firmware, I automatically tested the attenuation values of 0 dB, 2.5dB, and 12dB to see if there was any difference and found none. Attenuation is essentially signal boosting, and I was using a small jumper cable, so it made little difference. However, when swapping out the jumper cable for a short paper clip I found that the noise was reduced. When I get into designing PCBs I'm sure I'll understand this phenomenon better.

Raw ADC Values Testing



I tested ADC 2 with lower-level functions and found little improvement.

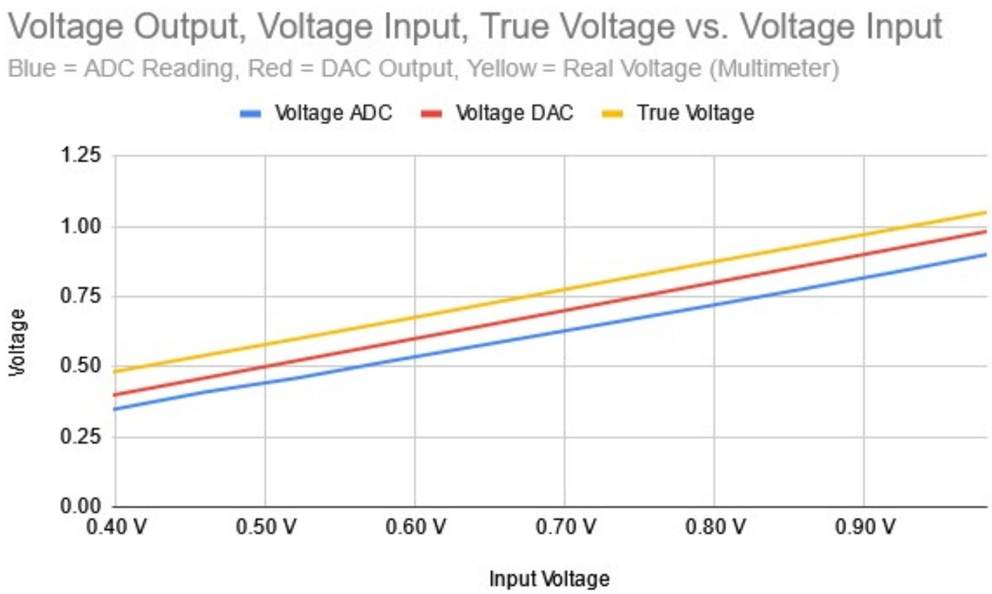
Looking further into the ESP32 documentation, I found lower-level libraries that can be used to get the raw ADC values directly instead of interfacing through the Arduino library. I wrote a quick function to convert this raw value into a voltage and tested it using the same "DAC voltage into ADC" strategy as before. I found no difference between my own custom conversion function and the Arduino library's analogRead() function. I also tested the second ADC on my ESP32 and found little difference in the results compared to the first ADC. However, the second ADC was accurate down to 0V while ADC 1 was accurate down to 0.08V.



With the graphs above, I got the error graph and fit a Polynomial (trendline) to it.

I then got the error for ADC 2 in raw bits. Instead of converting to voltages and then getting the error, I skipped that step and got the error in raw ADC values between 0-255. This isn't keeping with the ADC's 12-bit range because the DAC only goes to 8-bits, so I was limited to 8-bits of precision. I then fit a polynomial to the error and got the error graph above. The error was very similar to earlier testing, so nothing I had tried fixed the problem and mostly served as an exercise in understanding how to use the ADC on my ESP32.

Discovering the DAC was Imprescise



I tested the ADC and DAC with a multimeter and found that all the values differed.

During the previous round of testing with the raw ADC values I decided to take out a multimeter and get a ground truth value for the voltage. I discovered a discrepancy between the DAC's expected output voltage and the voltages the multimeter was showing. I tested it on a range of values between 0.4V and 1.0V and found a difference along the entire range - this is the data charted above.

Above you can see the True (multimeter) voltage, what the DAC thinks it's outputting, and what the ADC thinks the voltage is. All of these values differ by a significant amount (~0.5 - 0.1V). This means a self-calibration is not possible because the DAC's specified output and what it actually outputs are not equal.

Conclusion

To properly characterize and calibrate the ADC I would have to get the graph of ADC Value vs. True Voltage (As observed by a multimeter) and fit a polynomial to it in the range that the ADC error is mostly linear (0.1V - 3.0V). This is probably the solution to the issue we've had with the UBC Solar battery pack. However, if we find that the DAC on our chip is trustworthy, writing some firmware to automatically calibrate the ADCs on all of our boards would be a great way to ensure we never have this issue in the future on any boards.

This solution looks very obvious in retrospect, but it took a while to understand the systems and issues from a fundamental level to come to this conclusion with a high level of certainty. I've learned a lot about this project and there's even more to learn about why these ADC errors emerge in the first place.

Subscribe

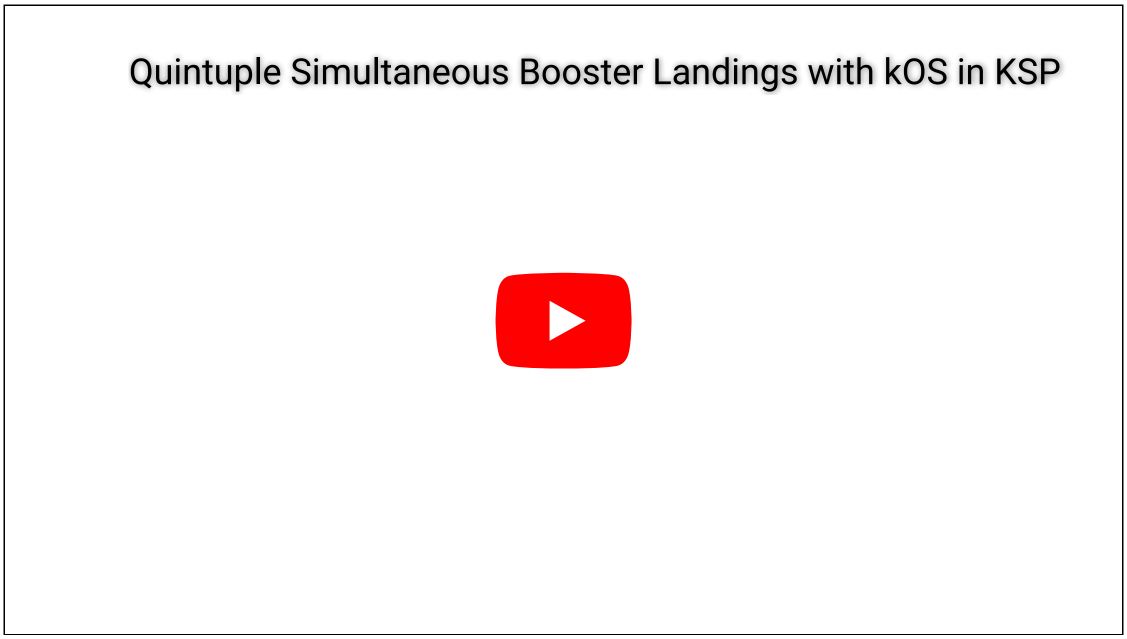
CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Quintuple Simultaneous Booster Landings with kOS

Dec 25, 2024 • Christopher Kalitin



Here's a video of the full mission

Here's a fun video from testing

X link if you have comments: here.

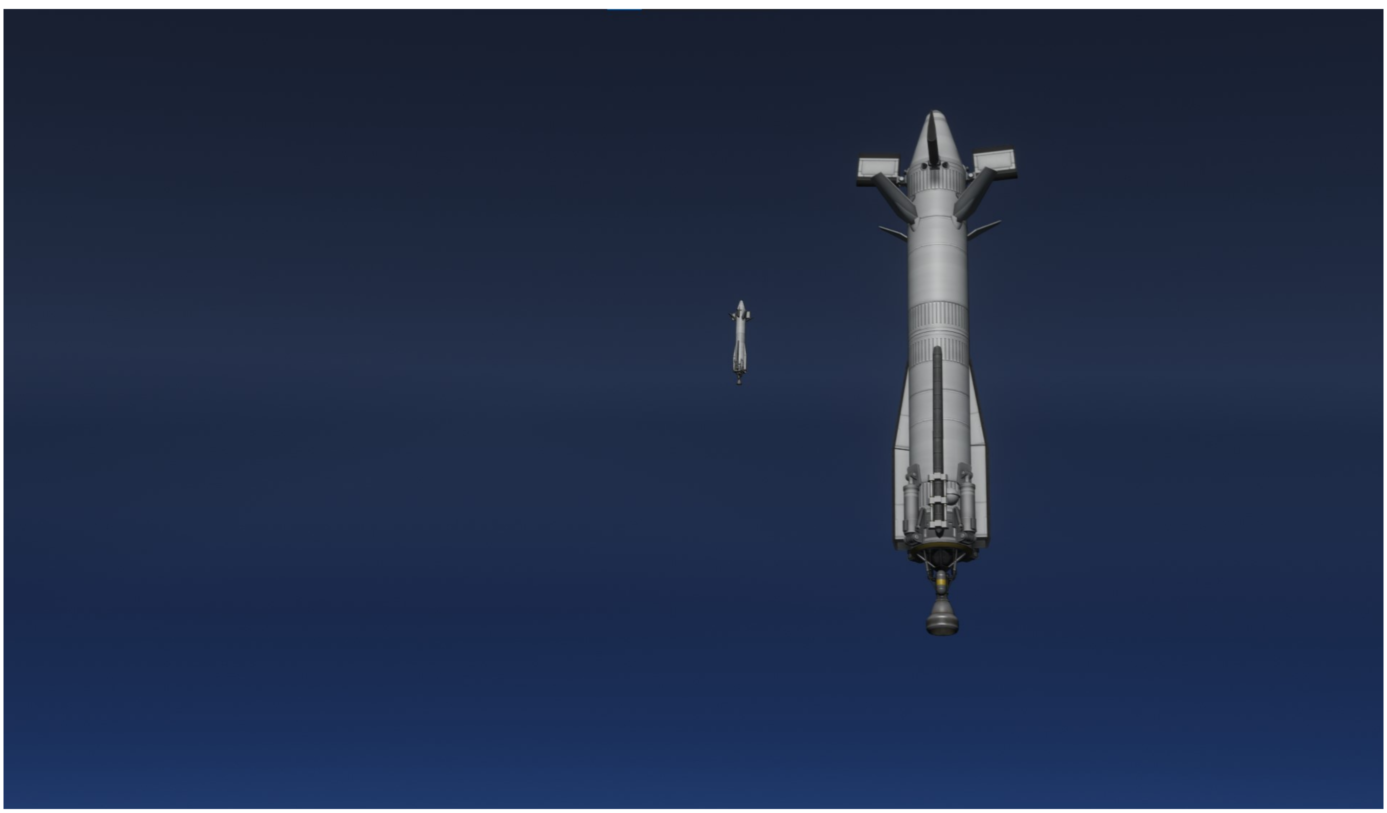
Read the code here.

I have a tendency to only write blog posts about things I think are impressive. This blog post isn't very technically impressive to me, but documenting your work has value in and of itself. Mischa Johal, THE UBC SOLAR WIZARD, has done a great job at drilling into our heads at UBC Solar the importance of documentation.

I was going to make this blog post understandable to non-technical non-kOS folk and hence increase the readership infinitely, however I realized I'm the only person who will ever read this.

This blog post is a continuation of my previous post on How to Land an Orbital Rocket Booster with kOS. In that post I described how I figured out how to write the fundamental code required to land a booster. Since then, I tinkered some more to create a fully autonomous KSP launch vehicle that can land 5 boosters simultaneously. It is completely beautiful and magical to watch agents you wrote autonomously navigate an environment - agents that are self landing rockets!!

kOS is still a shitty language



In my previous blog post I went on and on about why kOS is a shitty language. This remains true. The unique nature of this language makes it not very applicable to other projects. This has made me hesitant about continuing any projects with kOS as they aren't very technically informative.

However, rockets landing is magical. Since the previous blog post I found out there is a KSP mod for controlling rockets with Python. Rewriting the code with kRPC again falls into the category of a project that isn't very technologically insightful. Let's see if the lord gives me enough strength to work on firmware instead of KSP for the foreseeable future.

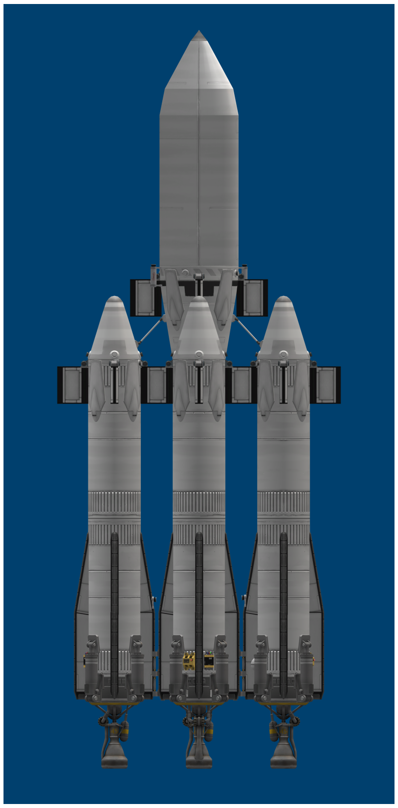
This is all to say that this isn't a technologically insightful or impressive project. There will be no impressive code, algorithms, or flight mechanics in this blog post. Just fucking around with a shitty language to get boosters to land. Interspersed with some remarkable pictures - Romans could have never imaged this!

Beauty is the Purpose

On real launch vehicles, getting the payload into orbit is the most important objective of every launch, recovery of hardware is secondary. However, unlike a real satellite, the output of my scripts is not economically productive. The output of my scripts is purely watching boosters autonomously land and marvelling at the sight. This is important context for understanding the parameters for state changes during launch (mass instead of velocity).

The Dzhanibekov (most impressive Cosmonaut to ever live who went on crazy Salyut repair missions) launch vehicle is a design that have a single core with 4 side mounted boosters. It has a payload capacity of about 1 ton in KSP. This is an extremely overengineered rocket because it's goal is to look cool, not be efficient. One productive insight from this project is a better appreciation of the difficulty of recovering high-energy stages. The side boosters separate at around 700/s and ~28km altitude. This is close enough to the launch site that the boostback burn is not very expensive. However, the center core separates at a far higher altitude and higher velocity, meaning it requires far more fuel to return to the launch site. This is sub-optimal as you are leaving a lot of payload capacity on the table. Hence why SpaceX doesn't try to recover the center core of the Falcon Heavy anymore.

How Ascent Works



Ascent is very simple.

Every stage on the rocket has it's own script and the center core controls the ascent. The center core script lerps between the initial pitch and the final pitch at the 25km, which is 45 degrees from vertical. These values were determined from my experience in KSP and flight testing. Also, note that I had to write the LERP function manually as kOS doesn't have it built it. Below is part of the function in the KOS-Scripts/Dzhanibekov/Dzhanibekov-Core.ks file. Like I said, no beautiful code in sight.

```
UNTIL SHIP:ALTITUDE > 30000 {
    SET targetBearing to 90.
    SET targetPitch to LERP(90, 45, CLAMP(SHIP:ALTITUDE, 0, 25000)/25000).
    SET targetRoll to LERP(-90, 0, CLAMP(SHIP:ALTITUDE, 0, 2000)/2000).
    LOCK STEERING TO HEADING(targetBearing, targetPitch, targetRoll).
}
```

Like all orbital rockets, we want to achieve a horizontal velocity of ~2200m/s to stay in orbit around Kerbin. However, we also need to return the boosters to the launch site. Because the boosters have to do boostback burns to arrest their horizontal velocity and move their impact location from far in the ocean to back at the same centre, we want to have as little horizontal velocity as is reasonable when we separate the boosters. That's why we use a lofted trajectory to get to orbit. For reference, as Matt Lowne explains in this video that taught me how to get into orbit in KSP years ago, on a standard ascent profile you aim for 45 degrees off vertical when you're at 10km. We aim for 45 degrees off vertical at 25km. This decreases the fuel required for the boostback burn and hence increases payload performance.

How Staging Works

The upper stage and boosters are dorment until their respective staging events. They wake up when particular mass values are reached. The core for this is shown below. Note that the UNTIL loop is just the opposite of a while loop. It runs the code until the condition is true.

Center Core Script:

```
IF (SHIP:MASS < 35.5 and SHIP:STAGENUM >= initialStageNum - 1) {
        LOCK STEERING TO srfPrograde.
        WAIT 2.5.
        STAGE.
        LOCK STEERING TO HEADING(targetBearing, targetPitch, -90).
        WAIT 0.5.
    }
```

Booster Script:

```
PRINT "WAITING FOR STAGING" at (0, 0).

UNTIL SHIP:MASS < 35.5 {}
WAIT 2.6.

Print "STAGED                 " at (0, 0).
```

```
LOCK THROTTLE TO 0.
```

The messaging system between separate scripts in kOS is slightly complicated, so I decided it was easier for each script to watch for staging events on their own instead of a central script telling all the others to wake up.

I tried to make each script detect staging events on their own, but this failed. The number of stages on a vehicle in kOS can be read with “SHIP:STAGENUM”. However, the number of stages on a vehicle only updates when the player is currently looking at a vehicle. This is a limitation of kOS as it has to work around KSP, which is designed for a single craft to be controlled at a time. This is why I had to use the “SHIP:MASS” value to detect staging events. The center core and boosters all detect the criteria for staging (Criteria: total rocket mass < 35.5t) individually and then move onto the next mission state themselves.

Batshit Crazy Boostback Burn Startup

The atmosphere of Kerbin is not balanced as well at God balanced Earth’s atmosphere. When we stage the boosters, we are still only 28km above the surface of Kerbin. This means aerodynamic affects play a major role in the control of the boosters. Unlike Falcon 9’s comparatively calm reorientation for boostback where it is above the atmosphere, our boosters are still in the atmosphere and have to fight against it. We can’t rely purely on RCS to reorient the boosters and atmospheric forces are too strong. So, we fire the engines once we are within 90 degree of the boostback burn direction vector. This leads to a batshit crazy looking separation and boostback startup as all the boosters are point in different directions and getting wildly flung around by the atmosphere. Watch the video at the top at the 1:51 mark to see what I mean.

Also, in testing I had the force of the decouple between the center core and the boosters set too high. Because the decoupler is at the bottom of the boosters for aesthetic reasons, this meant the top of the boosters would be pushed into the center core as the bottom was pushed away. Soyuz Style - This image really gets the point across. The solution was the decrease the force the decouplers exerted and to add small solid separation motors. Note that the force vector of the separation motors runs through the center of mass of the boosters to ensure no torque.

“Realistic” Upper Stage Burn

Once the center core propels the stack to an apogee of 77 km (exterimentally found to be the optimal value to have enough fuel to land), the second stage separates and continues onto orbit.

When you’re playing KSP yourself, the common way to get the second stage into orbit is to create a maneuver node at the apoapsis and burn prograde until you have your desired periapsis (likely above 70km, the barrier of the atmosphere). However, this usually leads to a coast phase between separation of the first stage and startup of the second stage engine. This is not very realistic, so I wanted to avoid it.

Instead, I start the second stage engine immediately at a throttle of 50%. The engine keeps running until we make orbit. This makes a far more realistic looking ascent profile.

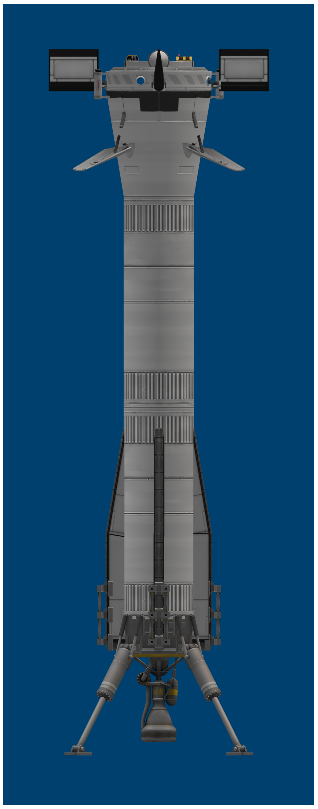
An issue emerges when the engine is ran at a throttle greater than ~50%. Because the second stage starts it's burn so early, it is still increasing in altitude and is far away from it's peak on it's parabolic flight, the apoapsis. This means that as we coast to apoapsis, the thrust from the engine is making that point higher and higher and further away in time. If this flight profile is followed, we keep burning to increase our apoapsis, but never efficiently enough to increase our periapsis so we reach a stable orbit (Oberth effect ftw).

To solve this, I have the second stage engine throttle down to 50% and control it's pitch to ensure the apoapsis doesn't run away from us. When we point down, the apoapsis decreases in height and comes closer to us in time. The opposite is true when we point up. So, the upper stage script is constantly adjusting the pitch to keep the apoapsis at a constant time away from us. It estimates the remaining time in the burn, and tries to make it so that we reach the apoapsis right when the burn ends. Once you understand the formulas, the math for this is very simple to implement.

```
CLEARSCREEN.
Print "BURNING TO ORBIT" at (0, 0).

LOCK STEERING to HEADING(targetBearing, 10, 0).
LOCK THROTTLE to 0.5.

UNTIL SHIP:ALTITUDE > 65000 { PRINT "WAITING FOR 65KM ALTITUDE FOR FINE CONTROL" at (0,

// We aim to burn until 10 seconds after the apoapsis to get into orbit, so calculate th
UNTIL ORBIT:PERIAPSIS > 75000 {
    SET shipdV to 9.81 * isp * ln(SHIP:MASS / SHIP:DRYMASS).
    SET remainingdVToLEO to leoVel - SHIP:VELOCITY:ORBIT:MAG.

    SET finalWetMass to SHIP:DRYMASS * 2.71828^(remainingdVToLEO / (9.81 * isp)).
    SET burnMass to SHIP:MASS - finalWetMass.
    SET burnTime to burnMass / massFlowRate.

    SET targetApTime to 10. // We want to be 10 seconds away from apoapsis forever

    SET timeToAp to ETA:APOAPSIS.

    SET targetPitch to CLAMP((targetApTime - timeToAp)*0.5, -30, 30).

    LOCK STEERING to HEADING(targetBearing, targetPitch, 0).
```

```
    PrintValue("Engine ISP", isp, 2).
    PrintValue("Engine Mass Flow Rate (t)", massFlowRate, 3).
    // The rest of the print statements are not included
}
```

Booster Aerodynamic Control Issues



When I was first testing the landing script, I only needed to test descent and not worry about ascent. This meant I only had to optimize for aerodynamics on descent. If you don't understand how to optimize a booster for descent I'm not gonna be able to explain it without being with you in person - If you'd like this please send me a dm. The basic principles are that you want high drag and for your center of mass to be below your center of drag. However, on ascent you want the opposite. The Falcon 9 solves this with deployable grid fins, but we don't have this in KSP 1 (RIP KSP 2).

Because your rocket burns fuel as it ascends, the center of mass shifts. We can set the rocket to first use fuel from the upper fuel tanks so that the COM shifts downwards. In a scenerio where our COL (Center of Lift) stays constant (Eg. New Glenn without deployable aero surfaces), at the begining of our first stage burn we could have COM below COL to ensure we fly point end up, and at the end when we're flying back, we could have COM above COL to ensure we fly engines first. In my first iteration of the vehicle this was what I aimed for with New Glenn / Superheavy chines at the bottom of the booster and non-deployable fins at the top, Howeverm the center of mass didn't shift enough to allow for stable ascent and stable descent.

The solution (as SpaceX learned on Falcon 9!) is deployable control surfaces. Instead of having static fins, I added joints from the Breaking Ground DLC to the fins so they could be stowed on ascent and deployed for descent. This solves to COL problem as we can artificially shift it when we descent by deploying fins at the top of the rocket.

Another issue was the drag on descent. When I was testing descent in the previous blog post, I created Falcon 9 Grass Hopper looking landing legs. These had a big base for supporting the legs that provided a tremendous amount of drag on descent. The completed rocket did not have this base and hence had far lower drag when on final descent. This is the difference between a terminal velocity of ~200m/s and ~500m/s. As you might imagine, this is an extreme difference in the fuel required to land. The solution is simple, air breaks. They look slightly ugly and are slightly unrealistic, but they work.

Another possible solution to this problem is to aggressively pitch the booster side to side on descent. This way, you can greater increase average drag force on the booster. Imagine this like doing S-curves while plummeting down to the surface. Although this would be a great mix between and extremely elegent simplification of the problem and a batshit insane looking descent, air breaks were far simpler to implement. Just add them to the craft and set them to an action group, simple.

Binary Search & Lack of Landing Pads

The Kerbal Konstructs mod (I think it's this one) I'm using to add the extra landing pads at the Kerbal Space Centre only adds three landing pads, presumably chosen for the required amount for a Falcon Heavy RTLS mission. I have 5 boosters so why don't we just they to get them to land at the same landing site in a pattern? Very beautiful, except I was slightly off when setting the coordinates for the landing position and they where perfectly aligned. Oh well.

The previous version of the code used the Trajectories mod to find the impact location of the boosters. However, the Trajectories mod is not designed to be used with multiple craft in kOS. This emerges as a phenomenon where the scripts do not get any impact position information if I am not actively focused on the craft. Worse yet, at state changes in the code, the scripts crash. So, a solution for finding the impact location had to be found that didn't use the trajectories mod.

Luckily, I had already created my own function for this that uses binary search to find the coordinates and time the booster would reach a particular altitude. KSP gives you information on your orbit at any given time, so with some bounds (eg. +0 and +10 minutes) and binary search, you can find the time in your orbit when you'll be at a particular altitude (eg. 0 meters). With the time you can convert that to a position using the same orbit information that KSP provides you.

However, the KSP Orbit information is all given relative to the core of the planet. This means that it doesn't account for the rotation of Kerbin. The simple solution to this is finding the circumference of Kerbin (2*600km*pi) and multiplying this by your eta to get an offset. This approach doesn't take into account aerodynamic forces (I think trajectories does this), but the error is decreases as we get closer to landing so the boosters asymptotically approach the correct impact location (the landing site).

Binary Search Function found in KOS-Scripts/HelperFunctions.ks:

```
// Return the lat/long of the position in the future on the current orbit at a given alt
// Ie. find the geolocation when we're at x meters above the surface in range y seconds
// SET impactGeoPos to GetLatLngAtAltitude(0, SHIP:OBT:ETA:PERIAPSIS, 10).
function GetLatLngAtAltitude {
    local parameter targetAltitude. // Meters
    local parameter timeRange. // Seconds
    local parameter altitudePrecision. // Allowable meters from given altitude to be cor

    // Replace 'SET' with 'Local'
    // Lower bound is present, upper bound is future
    local lowerBound to TIME:seconds.
    local upperBound to TIME:seconds + timeRange.
    local midTime to 0.

    // Binary Search
    for x in range(0, 35) {
        SET midTime to (lowerBound + upperBound) / 2.
        local midAltitude to body:altitudeof(positionat(SHIP, midTime)).

        if midAltitude < targetAltitude {
            SET upperBound to midTime.
        } else {
            SET lowerBound to midTime.
        }

        // If error less than precision
        if ABS(ABS(midAltitude) - targetAltitude) < altitudePrecision { BREAK. }
    }

    local geopos to BODY:GEOPOSITIONOF(positionat(SHIP, midTime)).
    // Longitude rotation of planet during coast to altitude ((360 degrees * seconds unt
    local rotationAdjustment to (360*(midTime-TIME:seconds)/BODY:rotationperiod) * cos(g

    return latlng(geopos:lat, geopos:lng - rotationAdjustment).
}
```

A Slightly More Unified Solution to Landing

In my previous blog post on kOS scripts, I wrote this:
"I imagine the solution is to track the estimated net displacement in landing position during the Suicide Burn. With the estimated time to touchdown, current pitch, and current horizontal velocity you could approximate the net displacement. Add this to the target landing location and it should be a much more accurate landing."

This quote is in reference to suboptimal solution to final propulsive descent that I came up with in my first iteration. An issue arises when your final landing burn is not perfectly vertical. Because there is a horizontal component to your thrust, your impact position shifts closer to you. If you were originally targetting your landing site before the landing burn, this horizontal component means you'll now land far short of it. My solution at the time was to adjust the target landing position by a constant so that we would initially overshoot, but end up landing right on target.

The addition of a constant was a very imprecise way to solve this issue. If we came it in different angles we would be off target (imagine, in the limit, a perfectly vertical or perfectly horizontal trajectory). So, the constant was just the eye-balled optimal value for an expected trajectory.

A better solution is to calculate an estimate of your net horizontal displacement during the landing burn. Then, you can add this value to your target landing location and end up far closer to the target in a wider range of trajectories.

```
function GetSuicideBurnNetDisplacementEstimate {
    local pitchRelativeToDown to vang(ship:facing:forevector, up:forevector). // Eg. up

    // Iterate over every second until impact and linearly estimate the angle relative t
    // With this value, calculate the difference in horizontal velocity, and add to net
    local localSuicideBurnLength to GetSuicideBurnLength().
    local t to localSuicideBurnLength.
    local netDisplacement to 0.
    UNTIL (t < 0) {
        local angle to lerp(0, pitchRelativeToDown, t / localSuicideBurnLength).
        local xVel to SIN(angle) * ((SHIP:AVAILABLETHRUST*EstThrottleInSuicideBurn) / SH
        SET netDisplacement to netDisplacement + xVel.
        SET t to t - 1.
    }

    return netDisplacement.
}
```

Above you can see the code for this estimated displacement function.

It works by getting your current pitch relative to vertical. It assumes you decrease your pitch linearly as you come in for landing. With this assumption, it steps through second by second to calculate your horizontal velocity at each step and takes the sum of these velocities to get your net displacement. This is a slightly shitty implementation and I didn't know the meaning of the word integral when I wrote this, but it works (mostly).

The "mostly works" part is adjusted for by another constant! Use this one easy trick to solve all your problems! Just add another terms! You can see the EstThrottleInSuicideBurn variable in the function above. We need to know our average throttle during the landing burn to get an accurate displacement result, and that's what this value represents. I set it to 1.5 (150%) in the config file for the boosters (KOS-Scripts/Dzhanibekov/DzhanibekovBoosterEast.ks). This is not quite a real value because we of course never expect more than 100% throttle, but I experimentally found it to be the proper value. A common theme in this project is experimentally finding proper parameters mostly because the experiments here are watching rockets land!

The config file for the boosters contains slightly more information than just the estimated throttle during the landing burn. It also contains the pitch multiplier that specifies how much the boosters should pitch to minimize error between intended landing site and impact position, craft height, and other variables. This "config" file isn't quite a config file because kOS doesn't allow this (shitty language). So, it also has the code to handle staging and running the landing script after the boosters have separated from the core.

I’ve covered the most important parts of this project, If you have questions, DM me.

MERRY CHRISTMAS!!! 🎄🎄🎄

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

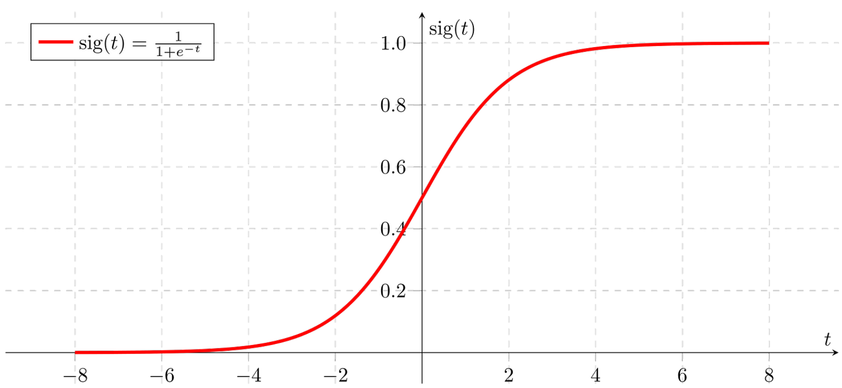
Christopher Kalitin Blog

S-Curves Allow You to Predict the Future

Nov 19, 2024 • Christopher Kalitin

I've decided I'm going to be an Econophysicist.

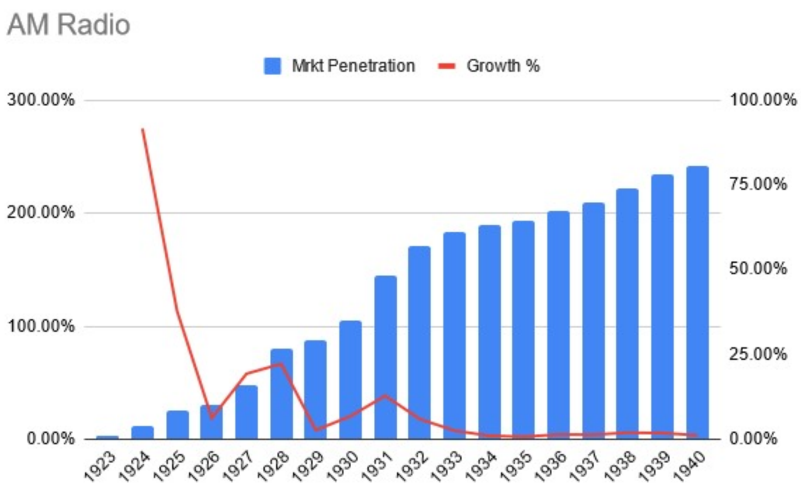
Historic Proof for S Curves

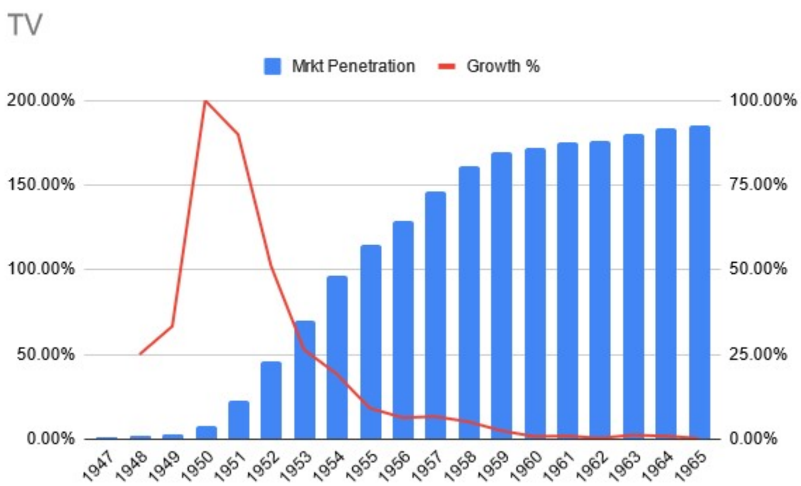


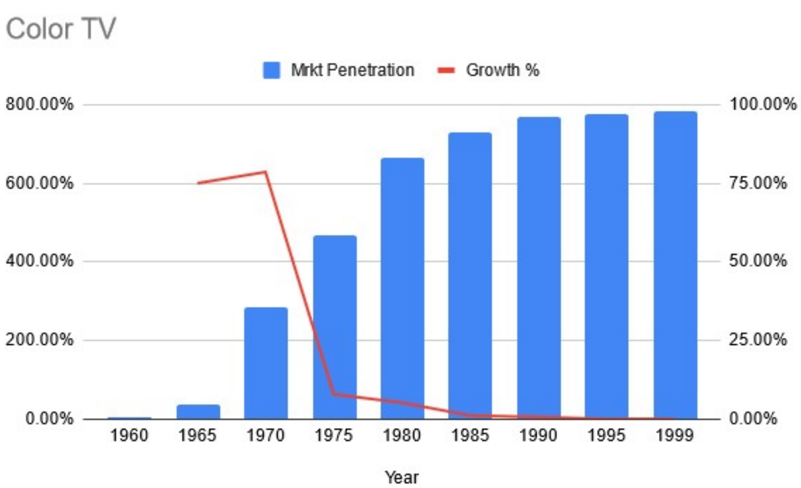
An S curve is a function that describes the shape of the market share vs. time graphs of almost all technologies. These S-curves are characterized by an extremely slow start, where they are arbitrarily close to zero; an exponential growth phase; then followed by a levelling off where they asymptotically approach 100% market share.

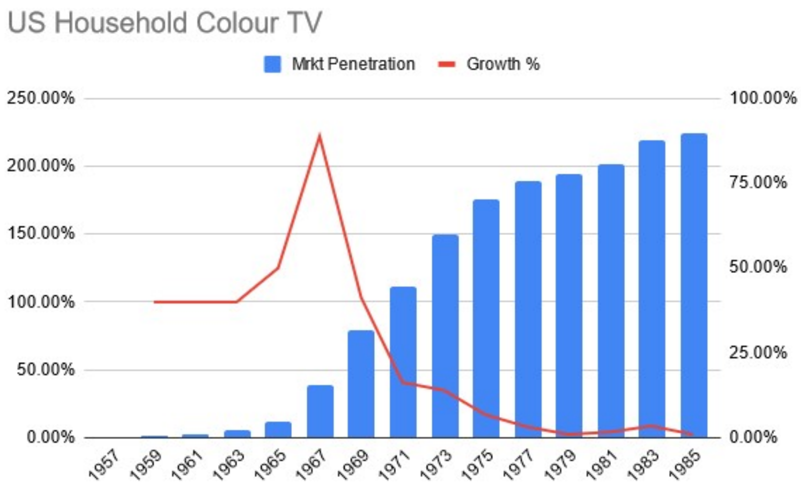
These S curves give you the ability to predict the future with a high level of certainty. Psychohistory brought to reality (Read Foundation). The reason you can use this method to predict the future with such high certainty is that the growth of almost all technologies that we have ever invented have followed the same pattern: the S curve.

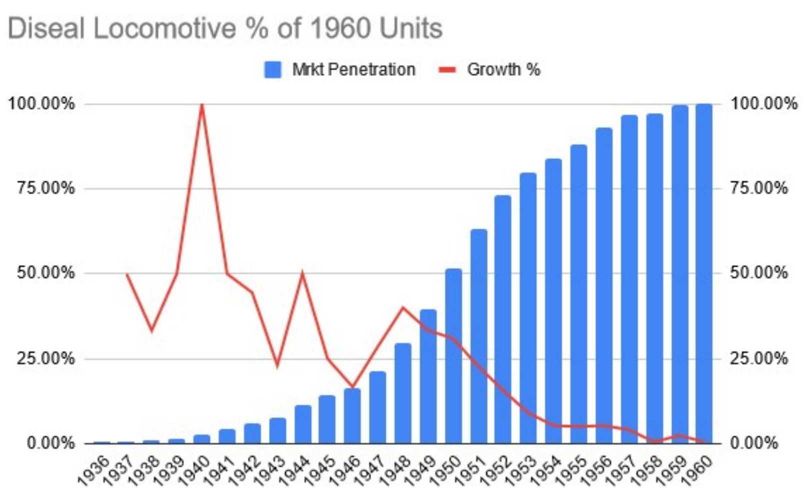
At the beginning of this year I began work on my longest blog post yet, The Transition to EV Robotaxis (FUTURE CHIRS! UPDATE THIS LINK WHEN ITS DONE). In that post I extrapolated historic data on EV adoption by fitting an S curve to the data. I found that 50% of cars sold in 2027 will be EVs and 90% in 2030. To ensure fitting an S curve was an appropriate method to make such predictions, I aggregated the market share data of several technologies over the past century to confirm whether they followed the same growth pattern. All of these charts are shown below.

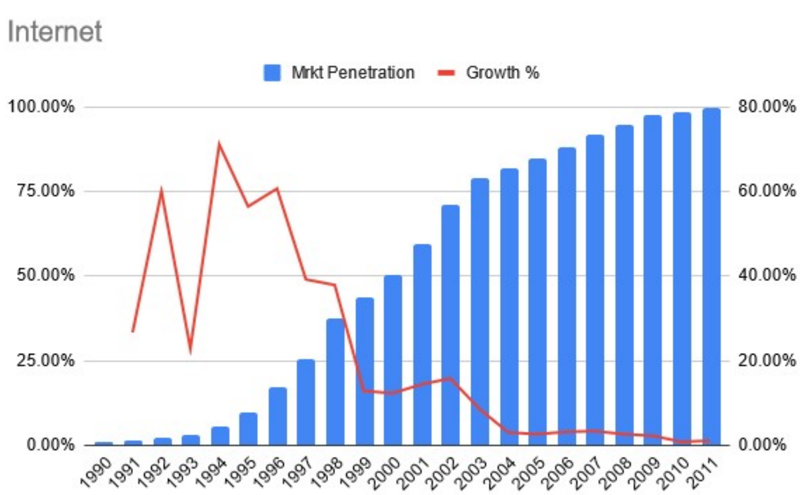

AM Radio
Mrkt Penetration
Growth %
300.00%
200.00%
100.00%
0.00%
100.00%
75.00%
50.00%
25.00%
0.00%
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940

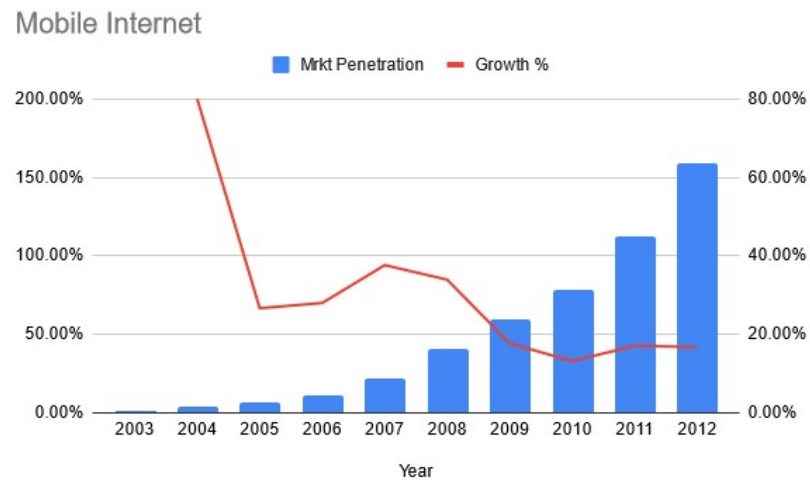

TV
Mrkt Penetration
Growth %
200.00%
150.00%
100.00%
50.00%
0.00%
100.00%
75.00%
50.00%
25.00%
0.00%
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965

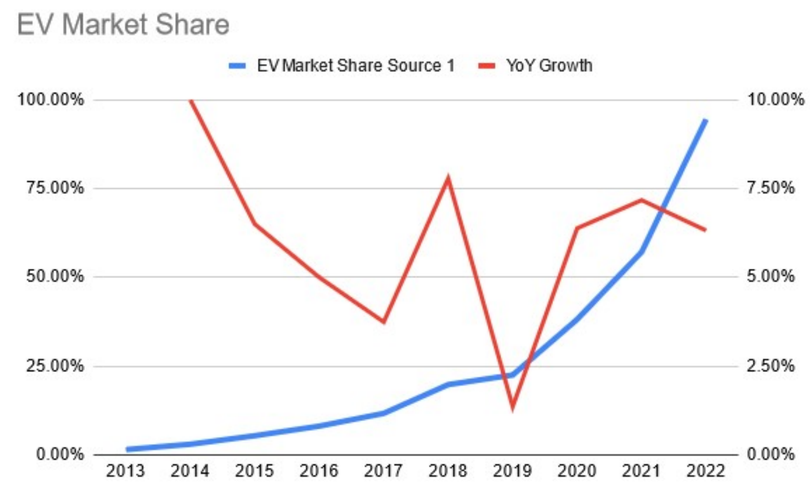

Color TV
Mrkt Penetration
Growth %
800.00%
600.00%
400.00%
200.00%
0.00%
100.00%
75.00%
50.00%
25.00%
0.00%
1960
1965
1970
1975
1980
1985
1990
1995
1999
Year


US Household Colour TV
Mrkt Penetration
Growth %
250.00%
200.00%
150.00%
100.00%
50.00%
0.00%
100.00%
75.00%
50.00%
25.00%
0.00%
1957 1959 1961 1963 1965 1967 1969 1971 1973 1975 1977 1979 1981 1983 1985


Diseal Locomotive % of 1960 Units
Mrkt Penetration
Growth %
100.00%
75.00%
50.00%
25.00%
0.00%
1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960


Internet
Mrkt Penetration
Growth %
100.00%
75.00%
50.00%
25.00%
0.00%
80.00%
60.00%
40.00%
20.00%
0.00%
1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011





Source of the charts

As you can see, all of these charts follow the same pattern. It’s important to note that I didn’t cherry pick any of these, I just searched for growth curves of early technologies and pixel counted to get the data you see above. Whether this introduces bias is up to you, but if you find any growth curves that don’t follow the S curve, please immediately send me a message. Such information would be extremely important - my preferred method of contact is carrier pigeon.

Learning Rate Explains it All

Assuming you aren’t frantically searching for a carrier pigeon to correct me right now, we can assume that all that historic data proves that all new technologies follow the S curve. If you have the mentality of a school student who only wants to answer the questions on the test, you can be satisfied with the information above and apply it to all new technologies to your heart’s content. However, if you’re destined for great things, you must understand why S curves happen from first principles.

As Casey Handmer (Highest information density speaker alive) explains in this clip and elsewhere, the fundamental reason that technologies follow S curves is the exponential growth that occurs when learning rate is allowed to compound.

Learning rate is the percentage decrease in cost that a technology experiences as a result of a cumulative doubling in production. For example, batteries currently have learning rate of about 20%. Batteries double in production about every 2 years and fall in cost by about 20%.

The framework of applying learning rate is most useful for new exponentially growing technologies where doubling time is low. For example, EVs have a doubling time of about 2-3 years. If we extrapolate this out a few years we find that we will undergo a massive paradigm shift where EVs become the most common vehicle type on roads. Electrical transformers can also be described by a learning rate, but because they are such a mature technology the learning rate is only 4% and doubling time is very long.

Like Handmer explains, exponential decrease in cost (described by learning rate) is due to the reinforcing cycle of increased demand which leads to increased production which leads to decreased cost which leads to increased demand. As a product increases in volume, the revenues from the product increase, which allows for further investment into R&D and production, which further decreases cost and increases the desirability of the product.

The most famous application of this framework is Moore's Law, which states that the number of transistors in an integrated circuit doubles about every two years. Moore discovered this with only about 4 points on the graph of transistor count vs. time. Because learning rate appears to be a fundamental law of the universe, he was able to predict the future of integrated circuits far into the future with a high level of certainty.

An even more remarkable fact about Moore's Law is that it applies to computers that existed well before Moore stated his law. When we look at the electromechanical, relay, and vacuum tube based computers that came before integrated circuits we find that the trend of exponential growth was present long before Moore's Law was discovered. Learning rate even works in the backwards direction before we ever thought about it!

The Life Cycle of a Technology

Precursor stage: dreaming, eg. Da Vinci drawing aeroplanes
Invention stage: the birth of a technology
Development stage: often more crucial than invention, many improvements that may be more important than the original invention
Maturity stage: interwoven into the fabric of life
Stage of the false pretenders: upstart threatens to replace the technology, but is missing some element, fails to surpass the original invention, short victory for the original technology
Final stage: new technology renders the original technology into a stage of obsolescence
Actual final stage: The complete end of the original invention, eg. horse and buggy, records

The Life Cycle of a Technology

We can identify seven distinct stages in the life cycle of a technology.

1. During the precursor stage, the prerequisites of a technology exist, and dreamers may contemplate these elements coming together. We do not, however, regard dreaming to be the same as inventing, even if the dreams are written down. Leonardo da Vinci drew convincing pictures of airplanes and automobiles, but he is not considered to have invented either.
2. The next stage, one highly celebrated in our culture, is invention, a very brief stage, similar in some respects to the process of birth after an extended period of labor. Here the inventor blends curiosity, scientific skills, determination, and usually of showmanship to combine methods in a new way and brings a new technology to life.
3. The next stage is development, during which the invention is protected and supported by doting guardians (who may include the original inventor). Often this stage is more crucial than invention and may involve additional creation that can have greater significance than the invention itself. Many tinkerers had constructed finely hand-tuned horseless carriages, but it was Henry Ford's innovation of mass production that enabled the automobile to take root and flourish.
4. The fourth stage is maturity. Although continuing to evolve, the technology now has a life of its own and has become an established part of the community. It may become to intertwined in the fabric of life that it appears to many observers that it will last forever. This creates an interesting drama when the next stage arrives, which I call the stage of the false pretenders.
5. Here an upstart threatens to eclipse the older technology. Its enthusiasts prematurely predict victory. While providing some distinct benefits, the newer technology is found on reflection to be lacking some key element of functionality or quality. When it indeed fails to dislodge the established order, the technology conservatives take this as evidence that the original approach will indeed live forever.
6. This is usually a short-lived victory for the aging technology. Shortly thereafter, another new technology typically does succeed in rendering the original technology to the stage of obsolescence. In this part of the life cycle, the technology lives out its senior years in gradual decline, its original purpose and functionality now subsumed by a more spry competitor.
7. In this stage, which may comprise 5 to 10 percent of a technology's life cycle, it finally yields to antiquity (as did the horse and buggy) the harpsichord, the vinyl record, and the manual typewriter).

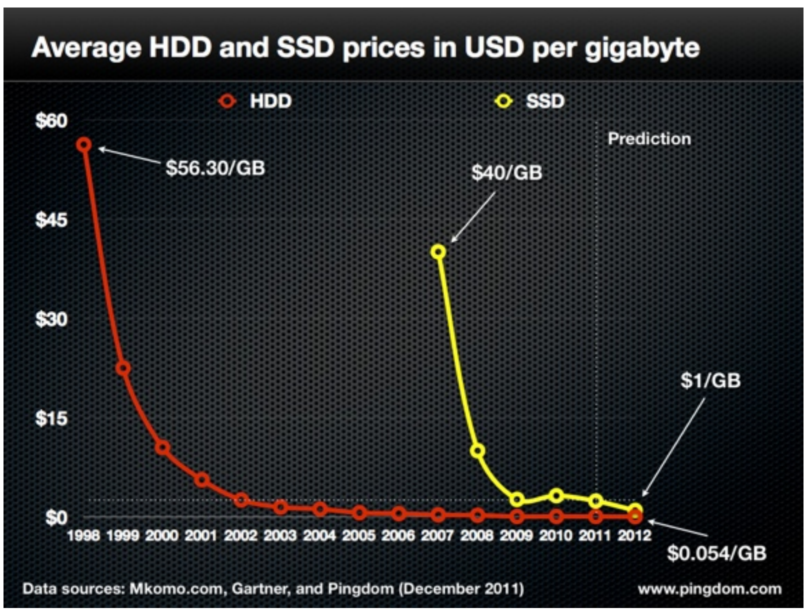
The Singularity Is Near, Chapter 2, page 105 on Internet Archive

Above you can see the seven stages that Ray Kurzweil laid out for the life cycle of a technology. The Singularity is Near was the first place I was exposed to these ideas presented in such a concrete manner. My first blog post was about insights I gained from this book, you may also want to read it.

A technology goes through five majors stages that can be described by its S-curve. These stages are:

1. Early R&D
2. Initial Commercial Appeal
3. Obvious exponential growth
4. Market Saturation
5. Stagnation

1. Early R&D

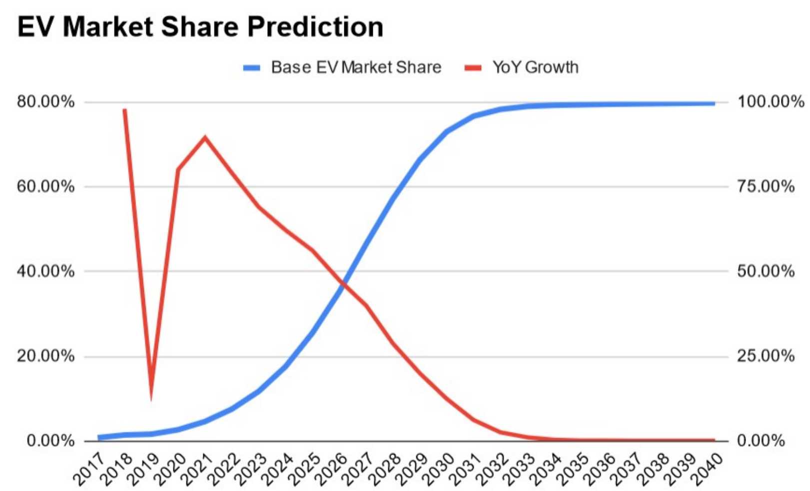


The early growth of a technology is characterized by exponential improvements that do not make much of an impact for its total market share. The technology may be exponentially improving, but it is still not good enough for anyone to use it at scale. For example, Solid State Memory (SSDs) were conceived of in 1978 and were first released as a product in 1991. However, it took until the late 2000s for costs to drop low enough that they were considered a reasonable alternative to hard drives.

2. Initial Commercial Appeal

After this initial R&D phase, there is a smooth transition to a phase in which the technology begins to be adopted. It has exponentially declined in cost enough that it has its own niche. An example here is golf carts, they are a relatively small and niche market compared to all other vehicles, but it just so happens that the electric powertrain is more optimal for the user experience than an internal combustion engine. This means that electric vehicle technology can be adopted to golf carts, which in the grand scheme of things isn't a very large market, but is just a step along the path to market domination.

3. Obvious exponential growth



Another gradual and smooth transition occurs from the second phase into the third, exponential growth. The technology has declined in cost and improved in capability enough that it is now the best option for a large portion of the market and it begins to take significant marketshare. We are living through this today with EVs. Pure electric vehicles currently have around 25% market share worldwide and the current growth rate is around 50%. Extrapolating this out a few years, we find that EVs will be the most common vehicle type on the road.

Because the exponential growth in this era is driven by economic factors (decreasing cost and increasing capability), the growth continues even through recessions and depressions. For example, look at the charts on AM radio during the Great Depression and EV adoption during Covid.

4. Market Saturation

No product can grow exponentially forever as we live in a finite universe. Once the market for a technology has been saturated, growth rates begin to decline. Now, unlike phase 2 where the technology only had its own niche, the technology makes everything else a niche. Horses were dominant before the internal combustion engine, but now are mainly used for niche recreational purposes, meanwhile the internal combustion engine powers the majority of vehicles.

It's important to note that growth rates linearly decline as a technology increases in maturity. This is again due to the fact that no technology can grow forever. You can see this in any of the charts above or my prediction for EV market share. Also, given a linearly declining growth rate, the magic of derivatives means that we get a nice smooth exponential looking curve. Next time you hear that the growth rate of EVs is declining, remember that this is a byproduct of market domination.

5. Stagnation
In the final stage of a technology, it asymptotically approaches 100% market share and we await another disruptive technology to emerge that displaces the current paradigm. We currently see this in internal combustion engine powered automobiles. We have had these products for over a century and there are very marginal improvements in their efficiency and cost. Most of the improvement we've seen in vehicles recently has been in comfort and other user-focused features.

The End of Learning Rate and the Death of a Technology

In the final stage of the life cycle of a technology, we see the end of learning rate and the slow death of the technology. As we learn more about how to harness a particular technology, we near the limits of the technology. This means that the once constant variable of learning rate begins to decline and the technology stagnates. Once you're in this stage, applying learning rate is no longer a very useful exercise to understand the future of the technology because the technology has very little future apart from what it already is. This is why learning rate is best applied in stages 2-4.

This stagnation is the time period in which we can hope to see a better technology arise that will supercede the current technology. In some cases, waiting for the emergence of this new technology can be a very long process that has extremely major consequences for the future of humanity. We are unable to predict when this new technology will emerge if it is still early in stage one - unless you have a perfect mental model of all similar technologies that are currently being researched and are able to predict their success.

Waiting on these new technologies can have major implications because of the impacts of the current technology. For example, Climate Change is caused by the previous/current paradigm of Hydrocarbon-based fuels. This technology was a huge win for humanity hundreds of years ago and allowed us to build the modern world and feed billions of people living better lives than they ever have. However, this technology has now begun to stagnate and we are seeing the negative impacts of it (citation needed). The solution to Climate Change is the next energy technology which will elevate humanity into a new era of unprecedented prosperity. Solar panels are batteries are declining in cost exponentially and will replace the vast majority of previous energy technologies in the coming decades. Modelling this out is left as an exercise to the reader.

The inherent hope of this perspective is that as long as humanity continues to develop technology, all of our problems will be solved and we will continually prosper to greater and greater levels. However, this isn't automatic! We must continue to work extremely hard to develop new technologies! The fundamental input into all technologies is human labour and we are that labour! You are that Labour! You have no clue how good the future is going to get when you work harder!

Appendix: More Examples of Applications of S-Curves

I'm currently reading *Eccentric Orbits: How a single man saved the world's largest satellite constallation from fiery destruction*. This book covers the Iridium satellite cellphone call constellation and it goes through a great example of an S-curve that looks like many discrete processes at first glance. The first stage began with the Motorola study into building a satellite constellation for handling phone calls began around 1988 (phase 1). The Iridium constallation began deployement in 1997 and the first call was made over the network in 1998 (Still phase 1). They had a brief encounter with bankruptcy and later began scaling the constellation and launching a new generation of satellites as revenue increased (Phase 2). At this point the Iridium phones were still a very niche product without much market penetration except in the very unique cases in which there was no better alternative. Next, Starlink entered the satellite internet market and began growing exponentially (Phase 3) (Note: I keep stubling upon this Brian Wang guy writing about SpaceX). We're currently living through phase 3 of the S-curve, in a decade or two we'll see Starlink reach market saturation and a few decades after that Starlink will be replaced by a new, better techonlogy.

If you imagine yourself at any stage in the story detailed above except the current one, it would not look like any exponential growth was occuring or like you could apply learning rate to understand the future of the industry - part of the reason for this is that little data would exist to determine learning rate or see the S-curve. You could be living through the brief bankruptcy of Iridium and think that the technology was a failure, but the underlying technology was still improving either through direct research or through external development. For example, while the original proposal for Iridium was being worked on at Motorola, satellite buses and other components were gradually becoming less expensive, and by extension the cost to develop a satellite internet/call constellation constellation was decreasing (Exponential Growth).

Appendix: Latent Demand

Latent demand is desire for a product or service that currently isn't offered in the market. Like I mentioned in the previous appendix, there are situations in which the fundamental technologies behind a product improve without direct investment into that product. For example, solar cells leverage semiconductor lithography which has been improving for decades due to investments into integrated curcuits. Over several decades, the fundamental technology behind solar cells was improving and decreasing in cost even while there was little direct investment into solar cells.

So, the theoretical price for solar was decreasing and hence the theoretical demand for solar was increasing. However, these prices had to be realized by increasing production through direct investment into solar cells. This means a technology can have a long period of indirect advancement, followed by a period of brief extreme growth that realizes the gains of the previous time period.

Another example is Starlink. There is a lot of latent demand for internet in rural areas and other locations that are not adequately served by current internet providers. However, there was no way to affordibly meet this demand. Existing companies like Iridium and all the others tried to meet this demand but could not do so. In comes SpaceX, which leveraged decades of materials science, electronics, and rocketry advancements to build the partially-reusable Falcon 9 which allowed them to launch the Starlink constellation at a low enough cost that they could provide affordable and high performance satellite internet service to underserved customers.

Throughout the decades before Starlink, satellite and launch vehicle technology was improving but these improvements were not being properly realized. Legacy launch companies like ULA had very little incentive to decrease cost and increase cadence. This never allowed an opportunity to get the exponential, compounding growth that learning rate describes. My blog post NASA's fucked - Here's my vision goes into some of the perverse incentives in the space industry that prevented this growth. In short, there was never a good enough opportunity for a company to create a satellite internet constellation before SpaceX came along. SpaceX realized decades of improvements in rocketery and pushed the field forward themselves to build the fundamental technology needed for a successful and profitable satellite internet constellation.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Why You Should Accumulate Capital

Oct 21, 2024 • Christopher Kalitin

What Money Is

This blog post is not "Why You Should Be Rich." When most people think about being rich the first thing that comes to mind is buying power. "I want to win the lottery so I can live on a yacht with a Lamborgini and an expensive watch." This is fundamentally spending your money to purchase goods and services - you do not get this spent money back. This spending is used to increase your quality of life or for other mainly hedonistic reasons.

Many self-made millionaire types (Real estate brokers / builders, sweaty startup founders, etc.) often call this hedonistic spending "burning capital." The transactions described above are not net positives for your bank balance. You spend money to receive goods that depreciate or for services that are consumed. For the purposes of this blog post, I will use this colloquial definition of money.

Fundamentally, prices are a mechanism for the allocation of scarce resources. You spend money to purchase these scarce goods/services. This basic economic principle is why spending money is a net negative for your bank balance.

For this next section, it's useful to temporarily remove the word money from your vocabulary. We need to solve the problem of allocating scarce resources. You require food, shelter, a mate, etc. to live and all these resources are scarce. This poses the problem of how best to allocate these resources to individuals. Under almost all economic systems you earn the right to scarce resources through providing value - even under Communism this basic meritocratic principle exists. You provide value through your work and are paid in a currency that can be exchanged for resources. For example, in feudal times you could work to produce wheat which is exchanged for other goods and services or given to your lord for the right to produce more wheat. Our modern economic system uses money as a measurement of both produced and consumed value. For this reason, in edge cases, it is possible to gain the right to scarce resources (eg. winning the lottery) without providing value (working).

What Capital Is

Money quantifies inputs into production processes and is often treated as an input itself. Just like all the other inputs - people, materials, land, etc. - it is consumed by an enterprise. However, the most important difference between money and other resources is that money is both an input and an output of all production processes. Unlike wood or steel, money doesn't constitute any physical object because money is a measurement of value. You cannot recoup the exact wood and steel that were used to build a factory, but you can recoup the exact money that was used to purchase those materials because every dollar is the same as every other dollar. Dollars are measurements, not anything physical.

Because money is fundamentally both an input and output, we don't need to consider it in the same way we consider other inputs. The most important element of capital is that it can gain value over time. Unlike money - which is spent and never returns to your bank account (mostly) - allocated capital can turn into more capital. An investor can put capital into a project and receive more capital in return - they receive the same exact resource they put in. This is not also true of lumber because lumber is consumed and not used as a currency. Fundamentally, money is spent on goods and services while capital is used to make more capital. Unlike money, capital is allocated, not spent. Put more simply, capital is money that turns into more money over time.

Capital is fundamentally productive because each production step adds value. If this isn't true, there is a negative return on invested capital. For any production done in a competitive free market, the value of the output must be greater than the value of the inputs. This is an axiom that applies to all production. If this axiom isn't satisfied, the market quickly corrects itself and either the producer adapts or goes bankrupt.

It should be clear why capital is more useful than money from the perspective of individuals (using the colloquial definitions described above). Capital makes more of itself and can be converted into money at any time by making purchases. This is fundamentally why investing works. Instead of spending your entire paycheque, you can invest in the S&P 500 and bet on the absolute productivity of those 500 companies increasing. Later, you can withdraw your capital and spend it on more goods and services than you could have purchased otherwise.

Capital Is Power

Allocate Capital to Issues You Care About

"A market economy allows accurate knowledge to be effective in influencing decision-making even if 99% of the population does not have that knowledge. In politics, however, the 99% who do not understand can create immediate political success for elected officials and for policies that will turn out in the end to be harmful to society as a whole."
Thomas Sowell, Basic Economics, Chapter 15

Under our current democracy, every voter has roughly even say in all issues. This is inefficient because not every voter has sufficient knowledge in every field to make an effective decision. Representative democracy tries to solve this by having the voter elect a representative to make decisions on their behalf. The reason this fails is that if you know less than someone in a particular field, you are not the best judge of their skill in that field. You cannot grade a math test if you know less than the students taking the test.

Instead of evenly allocating decision-making on all issues among all voters we can allocate decision-making to those who have capital allocated in a particular field. This solves the problem of having a lack of focus in a democracy where you have a say in all issues. There are likely only a small subset of issues that actually affect you and that you care about. The vast majority of decisions to be made in your country are likely uninteresting and irrelevant to you. If instead, you could allocate your votes to the particular issues you care about, you could have far more of an impact.

This system of allocating your vote to a particular issue exists today. If you want to steer a corporation in a particular direction, you can buy stock and vote as a shareholder. Anyone can acquire enough shares of any company to have more of a say in the management of that company than they do in the management of their country. For example, there are 161 million voters in the US. To acquire one 161 millionth of Tesla, you would need to spend ~$5,000. It's important to note that your interests may more closely align with a company of a smaller market cap, making your capital more powerful in that company.

Battling in the Free Market Of Ideas Leads to Better Results

"*The best way to predict the future is to create it.*"
Peter Drucker / Alan Kay

This system of allocating decision-making with capital invested in a particular field is even more powerful if you work in that field. Those who have the most power to steer a particular field in a certain direction are those who work in it. If you have your life savings invested in your field, you are likely to have a tremendous knowledge of the field and a great incentive to steer it in the direction you believe is right.

The participants in this kind of democracy are the most invested and most knowledgeable. They battle in the market to see which ideas are the best. Because this is an idea that gets to economic markets, economic principles decide who wins. By extension, the agents in a field that produce the most value are the ones that win. This is an extremely important point to understand because the winners in this battle of capital and ideas are those who will benefit the market - and by extension humanity - the most.

Capital is Meritocratic Allocation of Power Over The Future

"You may not be interested in politics, but politics is interested in you."

Capital allows you to participate in the debate over the future of humanity. What's more important than this is that you aren't being allowed to participate by some other power. You aren't at the behest of the government, which has a monopoly on violence and can forcibly take away your vote if suboptimal events transpire. Your capital is power, and you are the one who decides to enter the game.

This system of allocating power is fundamentally meritocratic. To accumulate a large amount of capital you must first invest a large amount of time, intelligence, and focus into your particular field. These are table stakes to enter the debate over the future of humanity. This is unlike a political class where you only get in the door if you kiss enough ass and backstab a critical mass of individuals. Your success in a free market is dependent on you, and by extension, your influence on the future of humanity is completely dependent on you. Under extremely political systems, power is allocated to those who have the right connections and not those who have demonstrated their ability to do the job. Stalin did not come to lead the Soviet Union by proving his skills in managing a nation.

Distributing Power Is a Moral Good

"The utopians who have dreamed of avoiding this struggle always land in some variant where they force others to perform it for them."
The Capitalist Manifesto, Chapter 2

Under capitalism, power is concentrated under the rich and under the political class. Under communism, there is only the political class. Concentrating power in the hands of a small number of people is how we get evil and suffering. For this reason, distributing power is extremely important to have a bright future for humanity.

The Free Market Allows Autonomy For All

"A government is just a corporation in the limit."
E

The defining feature of a free market is that market participants are free to make their own decisions about which transactions to partake in. If you apply this principle to a society, everyone is free from the control of anyone other individual or group - so long as we all have the power to decide if we do or do not want to participate in transactions. Because a true free market allows individuals to make their own decisions, it is not purely rule by capital. A problem arises when you consider how to maintain these individual freedoms and not degenerate into rule by capital, or, in the limit (Using the mathematic definition of a limit), rule by violence.

One of the beautiful manifestations of the mutual benefits of the free market is the "double thank you" as described in chapter 2 of The Capitalist Manifesto. Every time I've sold something on Facebook Marketplace both I and the buyer say thank you. We both agree to participate in a mutually beneficial transaction and both come out ahead. Thank you for allowing me to purchase this and thank you for taking it off my hands.

Liberal governments (in the classical sense of the word liberal: read What's Our Problem by Tim Urban and The Road to Serfdom by F.A. Hayek) have historically played the extremely important role of ensuring individuals have the freedom to make their own decisions. Violence is fundamentally required to ensure this fundamental freedom continues to exist because those with the power of violence have the ability to take away the freedom of others. For this reason, along with many others, the government has maintained its monopoly on violence within its own borders.

The issue arises when the government starts to abuse its absolute power of violence to impose its own will on the populace. Using capital to decentralize this power is a step in the right direction because it is far easier for a single entity with absolute power to control you rather than 1000 smaller entities. Individuals or groups with vast amounts of capital can also use this to control you, but from recent events, it is clear that the government exercises this power far more than private entities. The government is just a corporation in the limit. Would you rather have a single monopolistic corporation to transact with, or 1000 smaller corporations in competition where you can choose which one to transact with?

Concluding Thoughts

Get The Fucking Money

Chamath Palihapitiya did a great talk at Stanford in 2018, before the All-in Pod. This was the first place I was exposed to these ideas about gaining capital to be able to steer humanity into the path you see as the best possible future. The most memorable quote of his from this talk is, "Get the fucking money."

This is all on you. No one is going to implement your vision of the future for you. There's this idea from democracy that you can influence the future just by being given a vote. That is not true. There are vast currents in politics that you have very little control over by design - a tiny drop in a tsunami. You can either work hard to gain political power and become a member of this system that concentrates power or work hard to accumulate power and compete in the free market of ideas.

Capital Vs. Genius

td;lr

The point of this blog post was to illustrate how you can acquire capital to be able to build your vision of the future and why it is good for many decentralized individuals to all work towards building their own visions of the future. I'm mainly going into this theoretical idea and not how to actually implement it. How to acquire vast amounts of capital is left as an exercise to the reader. However, how to allocate this capital to maximize impact on the future is a theoretical topic that is important to mention in this context.

Capital efficiency is an extremely difficult metric to optimize for. You can't throw capital at this problem because it is fundamentally determined by the intelligence and ability of those involved in a particular enterprise. Companies that amass vast amounts of capital famously become less efficient at spending this capital. Government agencies also suffer from this inefficiency of complacency.

Consider Blue Origin and SpaceX. Blue Origin had more access to capital in its early days while SpaceX was far more limited. This is inferred from Jeff Bezos's higher net worth than Elon Musk at the time. SpaceX has obviously been far more capital-efficient and successful than Blue Origin. This is not purely due to the frugality of a startup with limited ability to raise capital, but also the genius of Elon Musk.

Human capital is what most often drives new advances forward. Once an idea is fleshed out and proven, capital can be thrown at it to scale until it takes over the world. It takes a Frtiz Haber to invent and prove the process to synthesise ammonia and a Carl Bosch with a massive corporation, BASF, to scale this technology to the point where it can feed the world. Many have tried to throw capital at reusable rockets and all failed, Elon Musk was the genius that succeeded. This was a situation where genius was far more important than capital. To have extreme impact on the future of humanity you have to be a genius, but we are not all geniuses so the best you can do is accumulate capital and allocate it to where you believe the geniuses are.

You Should Sit In The Sauna

The idea for the blog post came to be when I was half-dead in the sauna. My brain had completely shut down from the complete drain of energy that saunas impose on you, and out of this deep sleepy pit, I recovered my energy and pointed my mind in the direction of thinking about capital allocation. This is a beautiful meditative process and you need to do it. Completely shut down your brain and when it starts back up, point it in a specific direction. I imagine this is what Ayahuasca is like. I agree, I should take LSD.

One of the problems my recovering brain threw at me while I was still recovering my energy is how this philosophy of capital allocation connects with the idea of marriage. All your assets and money is shared in a marriage, which is a completely beautiful thing. But. My primary motivation in life is to maximize my impact on the future of humanity. Maybe it is wise for buying power (money) to be shared with my wife, and capital to be mine. Yeah, I should just read the Bible and go to church.

If you listen to economists for dozens of hours, you'll learn economics through Osmosis. I was halfway through Basic Economics by Thomas Sowell when I started writing this blog post. Now I'm reading The Capitalist Manifesto by Johan Norberg. Before this I read 1984 and The Road to Serfdom by Hayek. This was all through audiobooks I listened to (and took notes on) while on the bus or Skytrain in Vancouver. You can learn a tremendous amount just through diligent reading and taking notes. You should do this. You'll become a wizard in no time.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

How to Teach Engineering

Sep 15, 2024 • Christopher Kalitin

I've started university. I have a lot of ideas on university that'll probably evolve over my time here, might as well write them down in the early weeks to be able to look back on this in the coming years.

The Problem

I have an introduction to engineering class. I'm in a group of 5 and our purpose is to design a cardboard chair. We brainstorm cardboard chair ideas. We take quizzes on how to brainstorm. Then take more quizzes on how to pick which idea to take. Then finally after 4 weeks maybe we get to actually pick one of our own ideas. But it's very important that before that we play with paper for two hours and carefully observe how paper bends when we play with it. This is a requirement for playing with cardboard later in the class.

Who the actual fuck teaches engineering like this?

Ah yes make me wake up at 6 am every morning to take your quiz on how to play with cardboard. You haven't even given us cardboard yet, just paper. We have to work our way up to cardboard? really?

Make us write 3000 words over a month before you let us play with cardboard. I'm going to give my 2 year old cardboard out of spite. Watch him build a better chair at 3 than your months of lead up to giving 18 year olds cardboard.

The Real Process

I've had a tiny bit of experience with engineering projects.

This experience - and importantly comparing it to how my peers think about engineering - has illustrated to me exactly how useless this APSC 100 class is.

Engineering processes are emergent. You can't teach programming by focusing on syntax for two months before actually writing your own programs. Put ideas before implementation skills.

My intro to C programming class is only slightly better than intro to engineering. At least we build. It may be building a single function to output the value of x * y, but it's something. Afterwards you can show your classmates your 2 line solution versus their 10 lines and brag about how you're a better programmer. No one in the class has any idea what projects they want to make, but we at least get to build.

Out of this process of building, you start to understand why you actually do need to understand syntax and proper programming conventions. Then, you'll gladly read the textbook and learn the proper methods - you see the exact manifestations and efficiency gains by doing things the proper way.

Instead of this exciting process of exploration and figuring things out yourself - literally the process of science - you are forced to write 1000 words and with your group discuss the stakeholders for your cardboard chair.

The process of brainstorming -> evaluating -> building (and iterating back to any given step) is very useful. Let us figure that out ourselves.

The ideal of an engineer is someone who can turn dirt into value. How will you ever be able to do anything new if you're told everything that you must do by an authority?

A Beacon of Beauty in a Sea of Depressed Teenagers

I applied to the UBC Solar design team. Did my interview where I asked questions for 2 hours and answer questions for one hour. I was accepted.

By asking endless technical questions to the team I explored the space of ideas and drew my own insights. This process was the most fun I've had in months. Talking to techincal people and asking endless questions is the most fun and useful things in the world. I didn't fundamentally understand how engines or fuel cells worked before this week. By asking 2 hours of technical questions, I finally grokked it.

All kids are born explorers, it takes years - sometimes decades - for schools to beat this out of them. Science is the process of exploring the space of ideas and understanding the universe. This is fundamentally a self driven and explorative process. The way you get great engineers and scientists is by letting them practice this process, not by telling them to follow your arbitrary rules on how to build a chair.

The vast majority of students probably would have trouble if you simply said, "You have one month to build a cardboard chair that fits these criteria. Go do it." Wait where do we start? Brainstorm ideas? But I don't like my teammates ideas, how do we manage that? EXACTLY! How do you manage this? You will have to figure it out yourself and in doing this, you'll learn far more than being told what to do. ALL KIDS ARE BORN EXPLORERS. Let them explore and find out how to climb a tree without breaking bones.

Like I said above, if your process of planning how to build a chair is really the best way to do it, we will all reverse engineer our way into it and figure it out ourselves. Through this process we will understand a tremendous amount about how to execute on engineering projects.

Life is a Skill Issue

I only want "A Players"

As I type this I realize it may not be the wisest to categorize everyone into 3 buckets but this is how I believe we should look at everyone a part of the production team. You're either an A-Player, B-Player, or C-Player. There is only room in this company for A-Players. A-Players are obsessive, learn from mistakes, coachable, intelligent, don't make excuses, believe in Youtube, see the value of this company, and are the best in the goddamn world at their job. B-Players are new people that need to be trained into A-Players, and C-Players are just average employees. They don't suck but they arn't exceptional at what they do. They just exist, do whatever, and get a paycheck. They arn't obsessive

5

and learning. C-Players are poisonous and should be transitioned to a different company IMMEDIATELY. (It's okay we give everyone severance, they'll be fine).

Mr Beast's memo/book to new employees got leaked. Everyone has to read this.

Teaching processes makes C tier humans. Making your own processes breeds A tier engineers. God put me on this planet to build, and you're wasting my time by stopping me from exploring the space of ideas.

Everything in life is a skill issue and all skill issues can be solved by increasing your skills. That's why "Skill Issue" is such a powerful term, you are in complete control of whether your problems remain skill issues.

Some of my group members care more about grades than learning engineering. SKILL ISSUE C TIER BEHAVIOUR. You need to learn engineering yourself. It is fundamentally a self driven process of exploration. School isn't trying to help you, the members of the design teams would love to help you, but you need to ask the right questions!

This has been the first in a series of posts I'll probably do on University. My guess is that the principles behind my words will remain true for eternity. This is a one-shot blog post, I've been inspired by Mr Beasts constant spelling mistakes, just show your raw thoughts and yap. Eventually, your yapping will be intelligent enough that people want to listen.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Extrapolating Demand and Competition for the 1-ton Rocket Class

Aug 12, 2024 • Christopher Kalitin

Tell me where I'm wrong or just give compliments here.

Today four of the best space investing Youtubers posted their interview with Peter Beck and Adam Spice of Rocket Lab. Responding to one of Dave G's questions, Peter Beck said this about the 1-ton rocket class: "Our view of 1-ton is it's kind of a no man's land."

During this year I've written a few blog posts on the topic of 1-ton class rockets: Comparing Demand for Firefly's Alpha vs. Electron, Small Sat Constellations: The line between Electron and Rideshare, and Visualizing Small Sat Constellation Tradeoffs with Charts. The primary insights from this research have been that there is little competition between the 1-ton rockets and Electron and that currently there is only a small niche of the constellation market that is optimal for the 1-ton class. Peter Beck's comments help to confirm some of my conclusions and prompts more research into the small sat constellation market.

In this blog post, I'll use my dataset (Based on Jonathan Mcdowell's public data, Gunter's Space Page, and others) to quantify how big the market for the 1-ton class rockets is.

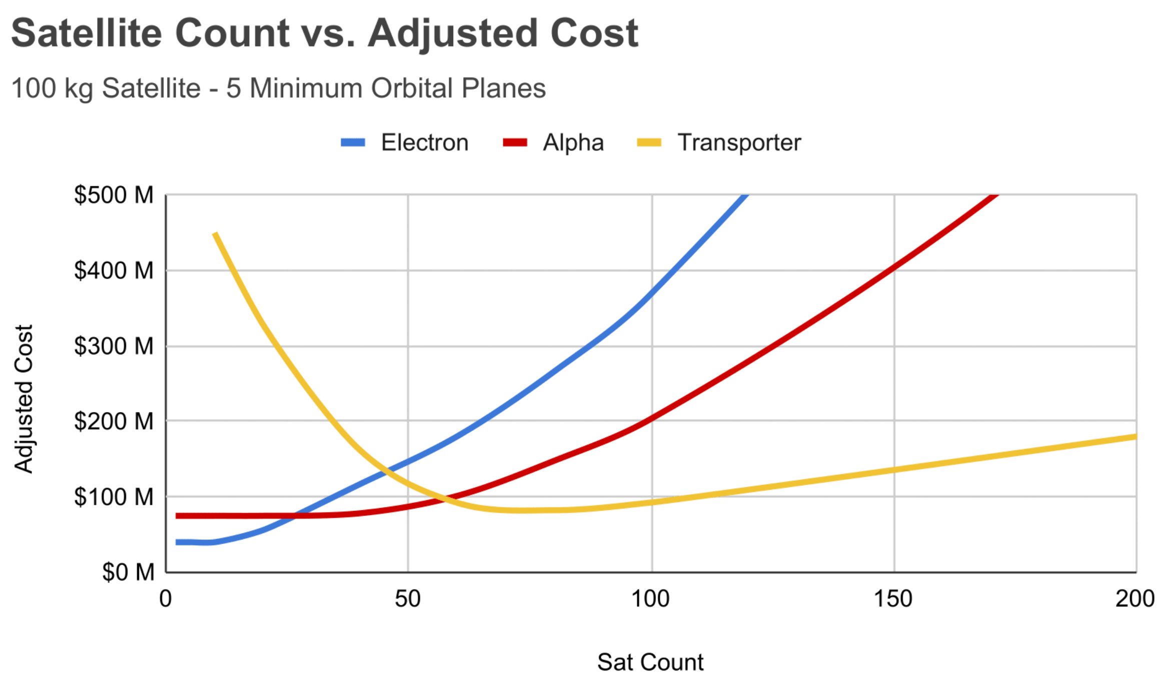
Don't Extrapolate the Early Launches

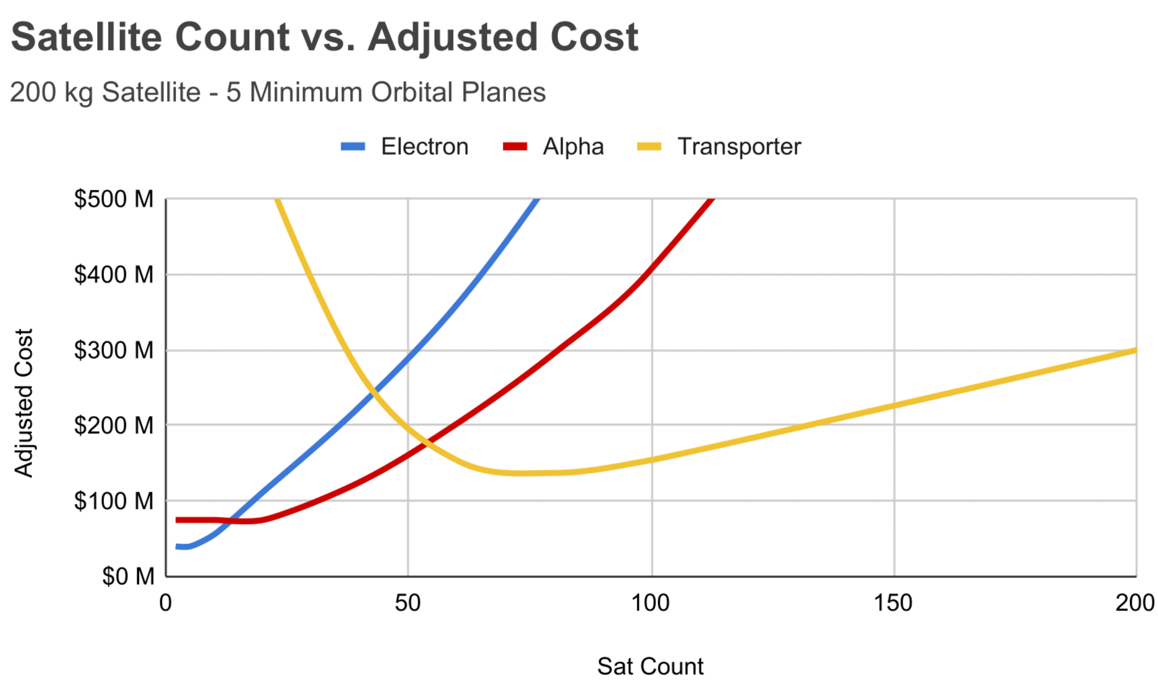
Despite the difference in size, Alpha's and Electron's first few flights resembled each other closely. Firefly's Alpha has flown 5 times to date and 3 of these missions were rideshare as part of their DREAM program or for NASA. This is the same as the first five Electron launches, 3 of them were rideshare carrying assorted satellites for Planet Labs, Spire Global, or NASA's ELaNa program.

As Alpha's and Electron's early launches proved, the first few launches of any vehicle are risky. This means that the first few launches of a new rocket are underpriced to account for the risk. The effect of this is the first few launches are not indicative of the future market due to the cheaper pricing. For example, in the last 3 years, Electron has launched 3 rideshare missions out of 28 total launches. "Beginning Of The Swarm", "Baby Come Back", and "There And Back Again". This amounts to ~10% of their launches vs. 60% during the first 5 launches. This is why we cannot directly extrapolate the first 5 Firefly launches and expect an accurate result.

"It's too small to be a useful rideshare vehicle and it's too big to be a dedicated vehicle." - Peter Beck. In this quote - again from the Dave G interview - Beck is referring to competing with Falcon 9 rideshare missions where specific orbit parameters and timing are worth sacrificing for a significantly lower cost. This reflects the early Alpha and Electron flights which were underpriced to account for the risk. However, Rideshare missions are not sustainable long-term on small launch vehicles because SpaceX has far better pricing power. In the second half of the quote, he says a 1-ton class rocket is too big to be a dedicated vehicle. This is because the vast majority of payloads that need the specific orbits and launch dates only available on dedicated launches are light enough to be launched on Electron, which is ~2x cheaper than Firefly's Alpha.

Most Constellations Are Optimized for Electron





The over-representation of rideshare missions is not the only source of uncertainty. The biggest upcoming market for rockets of all classes is constellations (70% of Rocket Labs 2024 launches so far). Small sat constellations are an even younger market than small sats themselves, so assumptions will have to be made in extrapolating the size of the market. The biggest assumption is what the distribution of satellite mass and total satellite count will be in upcoming constellations.

The competition between Alpha and Electron to launch constellations has parallels to the competition between Electron and Falcon 9 to launch small satellites. Electron fills the niche of payloads that either need unique orbits or an accelerated launch date. Because Falcon 9 is unable to properly serve these satellites, Electron can charge a higher price for a dedicated launch. Electron can't compete for the vast majority of small satellites launched by SpaceX on price, but they have a strong niche in the satellites that require their services. Alpha won't be able to compete for most Electron payloads because of cost, but they can create their own niche in the 1-ton class. The largest component of this niche is constellations that are optimized for 1-ton class launch vehicles.

In my previous blog post I created the beautiful charts you can see above that visualise the trade-offs between Electron, Alpha, and Rideshare in launching small sat constellations. Assuming you are using 100kg satellites, only when you exceed 25 satellites in your constellation does a rocket on the scale of Alpha become economical. Furthermore, the lower cadence of Alpha means Electron sees a further cost advantage.

The cost advantage of Electron starts to diminish when you increase the mass of the satellites. This is the strength of the 1-ton class rockets, they are the most efficient way to launch heavy and relatively small constellations. Satellites between 150-500kg in constellations of between ~25-100 total satellites are the optimal market for the 1-ton class rockets. The issue is none of these constellations currently exist, hence Peter Beck saying: "Our view of 1-ton is it's kind of a no man's land."

The heaviest commercial Earth Observation satellites are operated by Capella. Six of these 112kg satellites are currently in orbit and were launched on either Electron or Falcon 9 rideshare missions. These satellites are not heavy enough and not launched at a high enough scale to take advantage of Alpha's 1-ton payload capacity.

Are We Seeing a Military Constellation?

Lockheed Martin and L3 Harris have purchased up to 20 and 25 launches from Firefly respectively. It is not clear what payloads will fly on these missions and even if Lockheed Martin purchased flights on Alpha or MLV.

However, if launches are being purchased this far ahead in bulk it may be likely that these are for a potential constellation. The national security nature of these companies may also mean there is a responsive space aspect to these contracts that takes advantage of the capability Firefly developed for their Victus Nox mission.

Assuming these satellites are for a constellation, the choice of Firefly to launch them may be due to the fact that the satellites are heavy enough to take advantage of Alpha's payload capacity. Earth Observation satellites seem to level out at the ~100kg mark, but other applications may require heavier satellites. For example, various Starlink satellite iterations have been between ~250kg and ~800kg. If we are seeing the early stages of a military constellation, it may be that the hardware required for these satellites makes them fit into the class that can leverage the Alpha's 1-ton payload capacity.

The very forward-looking nature of these contracts may mean they don't materialize as This Guy (hilarious name) has theorised. Furthermore, my analysis is based on the data shown in the charts above. This doesn't take into account industry partnerships and other business factors.

The Two Categories of Satellites That Will Use Dedicated Small Launches

The first category of satellites that get their own dedicated small sat launch we've seen are those that use Electron for one of its unique capabilites. Compared to rideshare, Electron can launch to a specific orbit on a specific date. For commercial customers, this has meant a quicker time to revenue which should not be underestimated. Government customers have also taken advantage of the ability to launch to a specific orbit. For example in April they won a contract to launch a set of satellites for the US Space Force to a Very Low Earth Orbit (VLEO).

This category of payloads isn't very well addressed by Alpha because of the low mass of the payloads. For example, the USSF VLEO satellites weigh 200 kg in total. When payload capacity is not a concern, the cost advantage of Electron is a very large factor.

Peter Beck addressed this in his interview with Dave G: "If you look at the payloads that we're lifting they're all sort of in that 200kg class, so the whole reason someone comes to a dedicated rocket is because they want a dedicated rocket. So putting a 200 kg payload in a one-ton lift class, I just don't see how you can be economic to be able to compete with a smaller 300 kg lift class."

The second category of payloads we will see launch in the next few years are those that are too heavy for Electron and too light for Falcon 9. Just as Electron launches all satellites too light for Alpha, Alpha will be able to launch all satellites for light for Falcon 9. Because of the lack of competition and Falcon 9's utter dominance of the launch market during the last few years, it has launched several satellites to LEO that are far below its 18-ton payload capacity.

In order of most recent to least recent:

1. Korea Project 425 (800kg) (Anchor customer on Bandwagon-1)
2. iMECE (800kg)
3. EROS C3 (400kg)
4. Globalstar-2 FM15 (715 kg) (Rideshare with Starshield)

5. IXPE (330kg)
6. Paz (1,400kg)
7. Formosat 5 (525kg)
8. Jason-3 (553kg)
9. CASSIOPE (481kg) (First commercial Falcon 9 launch)

Given the 4x lower cost of an Alpha launch compared to a Falcon 9 launch, it is likely that Alpha will be able to compete for these payloads. Falcon 9 has launched less than one of these satellites per year over its lifetime, so the market is not large on SpaceX's scale. However, on the scale of Alpha, one launch per year is significant enough to mention.

Responsive Space

Sticking with the common theme of the 1-ton class filling niches, Responsive Space is a potential market for Firefly. Alpha's third launch was Victus Nox, a launch for the USSF that demonstrated its ability to launch a satellite within 24 hours of receiving the payload. To be able to launch a satellite so quickly, Firefly already had a vehicle at the launch site ready to receive the satellite and launch.

Rocket Lab was recently awardly a contract for the Victus Haze Tractically Responsive Space (TacRS) mission. This entails two launches, first True Anomaly's Jackal autonomous orbital vehicle, then a modified Photon spacecraft. The launch of the modified Photon will prove Rocket Lab's ability to launch in under 24 hours from receiving an order to and then maneuvering in orbit and performing rendezvous and proximity operations (RPO). Just like Firefly's Victus Nox mission, Rocket Lab will keep a vehicle in a "Hot Standby Phase" before the launch is ordered.

Awarding Rocket Lab this contract shows that the US Military is interested in multiple providers for responsive launches. Furthermore, tasking Rocket Lab with modifying a Photon spacecraft to perform RPO operations hints at the goals of creating responsive space assets. RPO allows you to shoot down enemy satellites. Firefly also builds Elytra, their orbital kick stage. We could see competition between the two companies for responsive space launches in the future.

Quantifying the Size of the Market

In the future, there will be four primary categories of launches for Firefly's Alpha, the most prolific 1-ton class rocket. These are constellations, dedicated large satellites, responsive space launches, and rideshare missions. The primary source of uncertainty in predicting the future of this market is the unknown requirements and goals of military customers. The military is understandably secretive about their plans and this makes it difficult to predict how many responsive space or constellation launches they will purchase in the future.

The current market Alpha enjoys with rideshare missions will not continue into the future because they are underpriced to account for the risk of early launches. These previous launches are underpriced in order to find customers for initial launches. With increased prices in the future, the primary reason to conduct this type of rideshare mission is an accelerated launch date compared to Falcon 9 rideshare missions. Since 2022, Electron has launched a single rideshare mission of this type.

Electron's other rideshare missions have either had unique requirements or were payloads tagging along with a larger primary payload. The recent "Beginning Of The Swarm" mission launched the NeonSAT-1 and ACS3 satellites to different orbits on a single mission by utilizing the Photon kick stage. About a year before this, Rocket Lab launched the "Baby Come Back" mission with the 30kg Teleset LEO 3 satellite and various smaller payloads for Spire Global and NASA. Firefly will be able to launch these types of missions with their Elytra kick stage and by ridesharing small satellites with larger primary payloads.

The market most well suited to Alpha is ~1-ton satellites. Judging from the earlier Falcon 9 missions, there may be one satellite per year that Alpha will be able to launch in this market segment.

The US Military has not made public how many responsive launches they will conduct in the future. This makes it difficult to predict the size of this market in the future. Given the competition betwene Firefly and Rocket Lab for these launches along with the developmental nature of these contracts, Firefly could see a responsive space mission every 1-2 years. Please argue with me if you have a differing view.

By far the market with the largest potential for Alpha is constellations. Currently, there are no constellations that have satellites massive enough and launched in enough numbers to take advantage of the payload advantage of Alpha, so we can only speculate on Lockheed Martin and L3 Harris' contracts and extrapolate from Electron. During the first 7 months of 2024, Rocket Lab conducted 7 constellation launches. According to Wikipedia, they have 23 planned constellation launches between 2024-2027. Given the fact that Electron will likely sign more contracts before 2027, the number of constellation launches over this timeframe should be 10-15 per year.

Earth Observation constellations don't appear to require Alpha's 1-ton payload capacity, but military constellations may be well suited for it. Lockheed Martin and L3 Harris both contracted Alpha for about 5 launches per year during a 4 year period. These periods have slight overlap, 2025-2029 and 2027-2031 respectively. We don't know what payloads are planned for these launches and during such a preliminary stage it is unlikely large sums of money have changed hands. For these reasons, the launch contracts should be discounted for the fact that they may not materialize. Around 3 constellation launches per year (Either military, large earth observation satellites, or other) is a reasonable estimate.

Conclusion

Summing the potential launches listed above, we get demand for about 6 launches per year. The breakdown is 3 Constellation, 1 Dedicated Satellite, 1 Responsive Space, and 1 Rideshare. Assuming an average price of $15M, this is a $90M market. Comparing this to ~20 $8M Electron launches per year, Firefly's TAM is about 50% of Electron's upcoming launch rate. This assumes that Firefly realizes the constellation market and that Electron launches don't substantially increase. Specifically looking at constellations - as they are the largest and fastest growing market - Firefly could launch 3 missions per year versus Electron's ~10-15. The higher price of an Alpha flight makes the gap smaller than the launch counts imply, 60% vs. 80%.

All the payload categories I've delved into above should make it clear that there is little overlap for competition between Electron and Alpha. Similar to Electron vs. Rideshare, Alpha is not cheap enough to take Electron's core market. Rather, it will create its own niche just as Electron has against Falcon 9 Rideshare.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

How to Land an Orbital Rocket Booster with kOS

Jul 24, 2024 • Christopher Kalitin

Tell me where I'm wrong or just give compliments here.

Read the code here.

One of the reasons Casey Handmer cites when telling people to write blogs is that they are proof of work. Many of my early projects will not be impressive at all, but it's worth documenting it for a few reasons. (1) To share my thought process throughout the project. (2) To force myself into documenting it. (3) To be able to reflect in the future. Forcing yourself to document something means it's much harder to be satisfied with the shitty way of solving a problem. This is why everyone should write a blog.

Can't think of a good way to write this, so I'll just detail all the mistakes I made and stupid things I did. You have to start somewhere!

I could write many thousands more words about this about the suicide burn, flight phases, printing, pitch mulitplier, and more. But all those things are less interesting and not what I want to cover. I'm not going to explain the simple stuff, this isn't a tutorial.

Initial Aero Control Approach

```
// Get distance between two positions without considering the altitude
// Eg. LatLngDist(V(SHIP:GEOPOSITION:LAT, SHIP:GEOPOSITION:LNG, 0), V(-0.09729775,-74.55
function LatLngDist {
    // Only x and y are used for lat/long. z is to be ignored
    Parameter pos1.
    Parameter pos2.

    // 10471.975 is the length of one degree lat/long on Kerbin. 3769911/360
    return (pos1 - pos2):MAG * 10471.975.
}

// Return direction to position in degrees starting from 0 at north
function DirToPos {
    // Only x and y are used for lat/long. z is to be ignored
    Parameter pos1.
    Parameter pos2.

    SET diff to pos2 - pos1.

    // atan2 resolves arctan ambiguity (ASTC quadrants)
    // Reversing x and y to rotate by 90 degrees so we start at 0 degrees at north, usua
    SET result to arcTan2(diff:X, diff:Y).

    // Keep degress between 0 and 360
    if result < 0 { SET result to result + 360. }

    return result.
}

// Return east/west and north/south components of velocity
function GetVelocityInCompassDirections {
    // https://www.reddit.com/r/Kos/comments/bwy79n/clarifications_on_shipvelocitysurfac
    SET vEast to vDot(ship:velocity:surface, ship:north:starvector).
    SET vNorth to vDot(ship:velocity:surface, ship:north:forevector).
    return v(vEast, vNorth, 0).
}
```

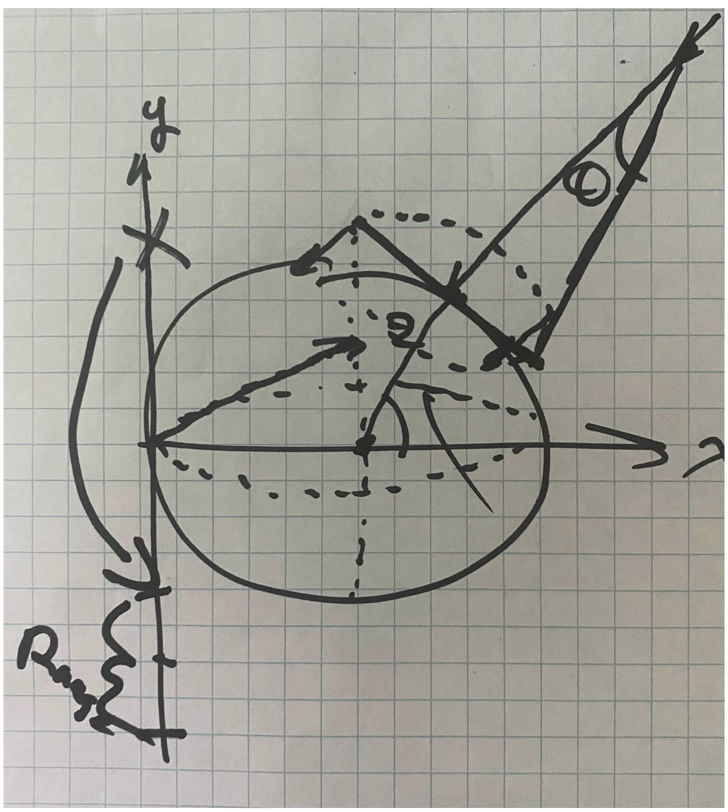
When I first attempted to land a booster with Kerbal Operating System about 2-3 years ago I got stuck trying to implement aerodynamic control. So, this is where I started.

In the intervening years I became a much better programmer and quickly implemented the helper functions above, kids stuff. The principle is to get the direction from the target point to the impact point and pitch in that direction depending on what the required change in distance per second is.

The horrible problem emerges when you realize that the DirToPos function returns the direction from one point to another on the surface of Kerbin. This value is then used as the bearing (degrees relative to north) for the booster.

Anyone who has attempted to land a booster in KSP knows that to adjust your landing site you have to adjust your pitch relative to the retrograde vector. The method above pitches relative to north (bearing relative to north, pitch is used as "amplitude"). At the scale I was testing at (~5000m above the Kerbal Space Center), this problem was not obvious. However, when attempting to reenter after a boostback burn the error became clear.

Failing to Clamp Angle Relative to Retrograde



In kOS (Kerbal Operating System, the scripting mod/language), you control the direction of the craft by inputting a heading which consists of a bearing and pitch value. This is easy to conceptualize for beginners as it's similar to the Nav Ball in KSP. However, what I didn't realize fast enough is that you don't want to do any operations on the heading. There's a reason why you use heading as a pilot and not when learning trig.

Imagine you're trying to clamp your booster's pitch to within 10 degrees of the retrograde vector. You only have the heading (bearing, pitch) value to work with. Pitch = 0 when straight up, bearing = 0 when pointing north. Stop and think how you would do it. Don't be lazy, do it.

Well if you're rocket is horizontal, clamping the raw bearing and pitch values will work great. A 10 degree offset in either bearing or pitch will result in a displacement of equal magnitude. However, if you're not horizontal (on the equator), a 10 degree change in bearing will result in a smaller displacement than a 10 degree change in pitch. This is the same reason why Vancouver is rotating around the Earth slower than Equador. Hopefully the diagram above makes this as clear as the diagram should be. If you increase pitch (closer to a pole), a single degree of bearing becomes shorter. When you're point straight up, a single degree of bearing is 0.

This problem took a few days to solve (because I don't know that much math, university solves this) and it all stemmed from the initial approach I used for heading control.

tl;dr this is very stupid:

```
function GlideToTarget {
    local aproxTimeRemaining to (SHIP:altitude - TargetPosAltituide) / (SHIP:velocity:su
    local targetChangeInDistanceToTargetPerSecond to impactToTargetDistance/aproxTimeRem

    // If impact dist < 50, do fine control that asymptotically approaches the target (b
    local pitchMultiplier to targetChangeInDistanceToTargetPerSecond * 2.
    if impactToTargetDistance < 50 { SET pitchMultiplier to (impactToTargetDistance^1.6)

    if RetrogradePitch > 70 AND ship:velocity:surface:mag < 450 { SET bearingLimit to 36
    else SET bearingLimit to pitchLimit.

    local shipDirToTarget to impactToTargetDir - 180 - RetrogradeBearing.
    local bearingAndPitch to GetBearingAndPitchFromDir(shipDirToTarget, pitchMultiplier)
    LOCK STEERING TO HEADING(bearingAndPitch:x, bearingAndPitch:y).

    // If retrograde pitch is nearing straight up, behaviour is not correct, so, lock to
    if RetrogradePitch > 80 {
        LOCK STEERING to HEADING(shipDirToTarget + RetrogradeBearing, pitchMultiplier).
    }
}
```

How to Properly Control Heading

Hopefully it's clear that bearing and pitch are horrible values that only pilots should ever use. In math class you use Eulers, Cartesians, etc. for a reason. The solution is to convert the bearing and pitch to a more suitable rotation system, clamp it, then convert back to bearing and pitch.

A very important insight I heard about learning to code is that half your time should be spent coding and the other half should be spent reading code. There are tons of people who've solved the problem you're working on, it is far more efficient to learn from them. This doesn't mean copy and pasting code, but truly understanding the problem and the solution.

I went back to the video that prompted me to try kOS again and found the solution. Turns out Donies did things the shitty way instead of truly learning (well everyone starts somewhere) and copied the code Edwin Robert wrote here (Video).

```
// I overengineered for 5 wasted days, this is the solution from: https://github.com/Dor
// This functions steers the ship relative to retrograde towards the target position, it
function GetSteeringRelativeToRetrograde {
    local Parameter pitchMultiplierLocalSteerRetrograde. // Local variable naming like t

    // Retrograde vector is in the SHIP-RAW Reference Frame https://ksp-kos.github.io/KC
    local retrogradeVector to -ship:velocity:surface.

    // :position converts from latlng to SHIP-RAW reference frame
    // Refactoring needed to minimize transforming values like LatLng
    local targetVector to ImpactPos:position - LATLNG(TargetPos:x, TargetPos:y):positior
    local targetDirection to retrogradeVector + targetVector * pitchMultiplierLocalSteer

    // If relative angle is too high, limit it.
    // Normalize the vectors, then multiply the target direction by the tan of pitch lim
    local angleDifference to vAng(targetDirection, retrogradeVector). // Angle of two ca
    if angleDifference > PitchLimit { SET targetDirection to retrogradeVector:normalizec

    return lookDirUp(targetDirection, facing:topvector).
}
```

Converting to the SHIP-RAW reference frame is the key. This allows for standard operations to be done on the rotation vectors and to use functions included in kOS like vAng.

The vAng function abstracts away some concepts I don't yet understand. Without it I would've had to study trig for a few weeks to properly implement it. Projects like this are great because they clearly show the extent of your knowledge. "retrogradeVector:normalized + targetDirection:normalized * tan(PitchLimit)." makes perfect sense to me and I can draw the diagram for you, but what goes on inside vAng is a mystery for now.

Failed Refactor

From the use of periods instead of semi-colons and other quirks like “local Parameter”, you might be able to tell that kOS is not a language meant for programmers, but rather for KSP players. This means a new mental framework is required to use kOS efficiently.

When implementing the intial helper functions and first attempt at landing I decided to use the techniques I was aware of and refactor in the future. This is immensely stupid and leads to a lot of wasted time that you will have to deal with in the future. Just write the code properly the first time.

The problem with the existing code is it used SET instead of LOCK on variables. In kOS you can declare a variable the way you’re used to by using SET. LOCK is used to update the value of a variable every physics tick. kOS is obviously supposed to be used with LOCK and my attempt at a refactor changed the code to use this different paradigm. The fundamental solution here is - of course - to rewrite kOS to be a proper language, but I ain’t doing that.

This refactor took more time than expected because I had to port the code to an entirely different execution paradigm.

Previous SET paradigm:

1. Create an infinite “UNTIL false” loop and keep a variable to track flightPhase.
2. Depending on the current flight phase, execute the appropriate function.
3. Break the loop when we’ve landed.

```
// directionError is SET in OrientForBoostback()

UNTIL false {
    if NOT ADDONS:TR:HASIMPACT { LOCK THROTTLE TO 0. CLEARSCREEN. BREAK. }

    UpdateFlightVariables().

    if flightPhase = 0 {
        PRINT "Flight Phase: Orient For Boostback (1/6)" at (0, 0).

        OrientForBoostback().

        PRINT "Flight Variables: " + numberHere at (0, 2).

        if directionError < 30 { StartBoostbackBurn(). }
    }
```

```
}

function StartBoostbackBurn {
    LOCK throttle to 1.

    SET flightPhase to 1.
    CLEARSCREEN.
}
```

New LOCK paradigm:

1. Lock global variables that are needed very often.
2. Call the first flight function (OrientForBoostbackBurn()).
3. Inside OrientForBoostbackBurn(), lock the appropriate variables and wait for completion condition.
4. When the completion condition is met, call the next function.

```
LOCK ImpactToTargetDir to DirToPoint(ImpactPos, TargetPos).

OrientForBoostbackBurn().

function OrientForBoostbackBurn {
    LOCK TargetHeading to Heading(ImpactToTargetDir, 0).

    WAIT UNTIL vAng(targetHeading:vector, ship:facing:vector) < 30 { BoostbackBurn(). }
}
```

The code above was my first attempt at the refactor. It failed because when WAIT UNTIL is called, all other execution stops. In the previous SET paradigm, I used WAIT(0.1) to control the tick speed (Which itself is flawed because the code needs time to run, so Hertz is actually <10). In the new LOCK paradigm, "WAIT UNTIL(completion condition)" simply pauses the program until the condition is met, which is never because it is never updated. Apart from this glaring flaw, this approach also doesn't allow printing variables continuously.

The solution is to add a loop (eg. 10Hz) to the end of the flight functions with this line: "WHEN (completion condition) { RunNextFlightFunction(). }". The loop can also be used to print variables continuously or kOS GUI widgets can be used (better and proper). This will also make the code far more readable. I would've done this but I was on week 3 and wanted to finish the project, maybe will in the future when bored.

No Unifed Solution To Cancel Horizontal Velocity and Minimize Landing Error

In the second part of the video at the top of this post, you can see the booster landing with the UI active. Unlike the first cinematic landing, this one barely makes it onto the landing pad.

The approach I implemented to have a soft touchdown has two phases. First, the Suicide Burn is started and it targets a point ~30 meters away from the landing pad in the opposite direction of the rocket. Second, when the rocket is <40 m/s or <25m altitude, the SoftTouchdown() function is called. This cancels out horizontal velocity and slowly decreases vertical velocity until touchdown (lerp between 10 m/s to 2 m/s, t=altitude/50). While the suicide burn is performed, the horizontal velocity changes and by extension the landing location. This is why aiming ~30 meters off is necessary in the beginning (shitty solution).

```
// Extra code not included, this gets the point across
function StartSuicideBurn {
    local magnitude to -(GetHorizationVelocity():mag^1.67) / 45. // Offset by multiple c
    SET TargetPos to AddMetersToGeoPos(targetSite, GetOffsetPosFromTargetSite(magnitude)
}

function SoftTouchdown {
    local t to TrueAltitude / 50.
    SET TargetVerticalVelocity to Lerp(-2, -10, CLAMP(t, 0, 1)).

    local aproxTimeRemaining to (TrueAltitude - TargetPosAltitude) / (SHIP:velocity:surf
    SET aproxTimeRemaining to CLAMP(aproxTimeRemaining, 5, 10). // Clamp to 10 seconds,

    local pitchMultiplier to Lerp(0, pitchLimit, CLAMP(GetHorizationVelocity():MAG/3, 0,
    LOCK STEERING TO HEADING(RetrogradeBearing, 90 - pitchMultiplier, 0).

    local baseThrottle to SHIP:Mass/(SHIP:MAXTHRUST / 9.964016384)-0.02. // Hover, Kn tc

    local vertVelError to TargetVerticalVelocity - GetVerticalVelocity().
    local throttleChange to CLAMP(vertVelError^1.7/50, 0.01, 0.25) * (vertVelError/ABS(v

    LOCK throttle to CLAMP(baseThrottle + throttleChange, 0, 1).
}
```

This approach has a poor success rate. With our two data points in the video, only 50% make it to the inner circle of the landing pad. A unified solution that both cancels horizontal velocity and minimizes landing error is needed.

There is a shitty solution here that many people have used. You can do an entry burn to cancel horizontal velocity far above the landing site, then land. This approach is shitty because it's unrealistic, uses extra fuel, and is avoiding a really fun problem.

I imagine the solution is to track the estimated net displacement in landing position during the Suicide Burn. With the estimated time to touchdown, current pitch, and current horizontal velocity you could approximate the net displacement. Add this to the target landing location and it should be a much more accurate landing. However, even this is a slightly shitty solution, maybe Rafael (Best kOS landing script I've ever seen) knows the right way.

The Fundamental Insight

The fundamental insight I learned in the past month of doing this project and watching Deep Learning lectures is that to build things properly you need sufficient knowledge of the underlying fields. The most important insights are often the most obvious ones, truly understanding and applying them is the important step. This project could have taken a few days if I was good at rotation/vector math and knew more about GNC. Skill acquisition is the most important goal all young people must have. After you've acquired the skills, building becomes exponentially easier.

"I may not be able to do it the good way, but I sure can do it the shitty way." Don't be a lazy fuck, you won't be a good programmer (or actually good at anything) by doing things the shitty way.

If I had good skills, I would've written the code like this (Video). I have a ton of respect for the man that wrote that code. He didn't do it the shitty way.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

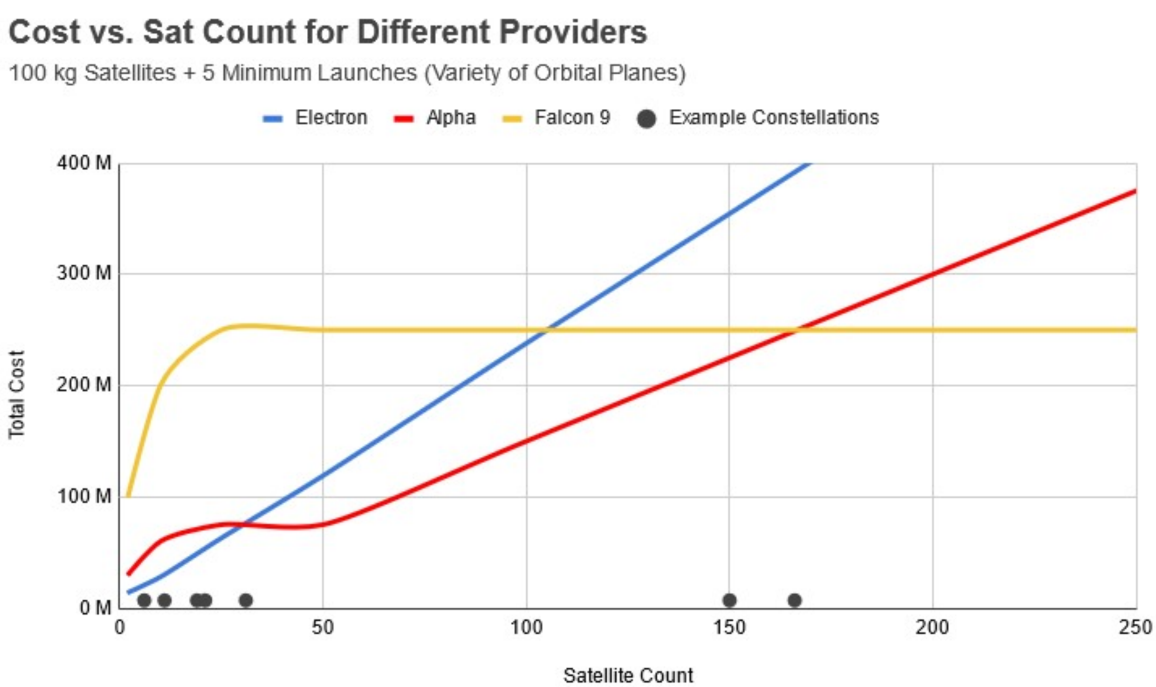
Visualizing Small Sat Constellation Tradeoffs with Charts

Jul 16, 2024 • Christopher Kalitin

Spreadsheet Link

Tell me where I'm wrong or just give compliments here.

What Data Do We Need?



In my previous blog post I gained an intuitive understanding of the tradeoffs for the different methods of launching small sat constellations. This blog post is an exercise in quantifying and visualizing that understanding.

Yesterday I felt the need to create a visual representation of the tradeoff between small sat constellation operators choosing either Electron, Alpha, or Transporter. The result was the chart you see above. It clearly shows the size of constellation where Electron, Alpha, or Rideshare are the best options. However, this doesn't take into account variables other than the raw cost, such as the suboptimal orbits of rideshare missions, true Transporter mission costs, or the low annual mass to orbit capability of Electron because of its "low" cadence coupled with it's low payload capacity (Isn't it great we get to live in a world in which 20 launches per year is "low" cadence, get on SpaceX's level guys!).

The solution to getting a more accurate picture of the tradeoffs is to abandon the true cost of launch and replace it with Adjusted Cost, a score that takes into account the issues laid out above. This is a simple concept, adjust the cost with a multiplier that uses the number of satellites as the input variable.

Orbit Detriment Multiplier (Rideshare): y = 85 * 2.71^(-0.08*sat count)
Cadence Detriment Multiplier (Dedicated): y = 0.0005(sat count - 20)^1.5

Orbit Detriment starts very high at 0 satellites and approaches zero as we near 100 satellites. This represents the fact that rideshare missions can't be used to launch small constellations that require specific orbits like NASA Tropics, Capella, or BlackSky. It shifts the calculus towards dedicated launches for constellations with a low number of satellites.

Cadence Detriment begins at 20 and scales exponentially with the number of satellites, this takes into account the fact that Electron can't launch 100 times on a whim. Furthermore, there is a minimum orbital plane input which sets the floor for the number of launches, this is required to get accurate data between Electron and Firefly's Alpha at low satellite counts.

The result is several charts that are available here.

Insights

1. Most Constellations Below 100 Satellites Are Either 100kg or 50kg

Data ends Dec 31 2023 (mostly)

Constellation / Company	Category	Sat Count	Launches	Sat Mass	Launch Vehicle	Orbits	Rideshare Addressable	Unique Plane Required	Planned Sats	Notes
Planet Labs	Observation	542	29	5.0 kg	Soyuz, Antares, I	LEO / Rideshare SSO	Yes (Scale)	No (Scale)	543+	
Spire Global	Communications	166	34	4.0 kg	PSLV, Atlas V, A	LEO	Yes (Scale)	No (Scale)	166+	Many uses: Ship tracking, m
Swarm	Communications	150	13	0.3 kg	Vega, Electron, F	Unique Plane / LEO / S	Yes (Scale)	No (Scale)	320*	Acquired by SpaceX
ICEYE	SAR	31	12	85.0 kg	Soyuz, Falcon 9	Rideshare SSO	Yes	No (Intermittent Data)	31+	Entirely F9
HawkEye 360	RF Situational Awar	21	7	30.0 kg	Falcon 9, Electro	Rideshare SSO / Bandw	Yes	No (Intermittent Data)	~33	Lately only F9
Black Sky	Observation	19	12	56.0 kg	PSLV, Falcon 9, I	LEO	No (Lack of Scale)	Yes (Lack of Scale)	60*	Lately only Electron, old sour
Capella	SAR	11	11	112.0 kg	Electron, Falcon	LEO / Rideshare SSO	No (Lack of Scale)	Yes (Lack of Scale)	30	
Fleet Space Centauri	Communications	6	6	35.0 kg	Falcon 9, PSLV, I	Rideshare SSO	Yes (Scale)	No (Scale)	140	Mainly Tech Demos so far
Kineis	Communications	5	1	30.0 kg	Electron	High SSO	No (Lack of Scale)	Yes (Lack of Scale)	25	
Tropics	Observation	4	2	5.3 kg	Electron	Unique Plane	No	Yes	N/A	
iQPS	SAR	4	4	100.0 kg	Epsilon, Electron	Rideshare SSO	No (Lack of Scale)	Yes (Lack of Scale)	36	Mainly Tech Demos so far
Synspective Strix	SAR	3	3	100.0 kg	Electron	Unique Plane SSO	No (Lack of Scale)	Yes (Lack of Scale)	25	
Prefire	Observation	2	2	15.0 kg	Electron	Unique Plane	No	Yes	N/A	

Expanded Chart

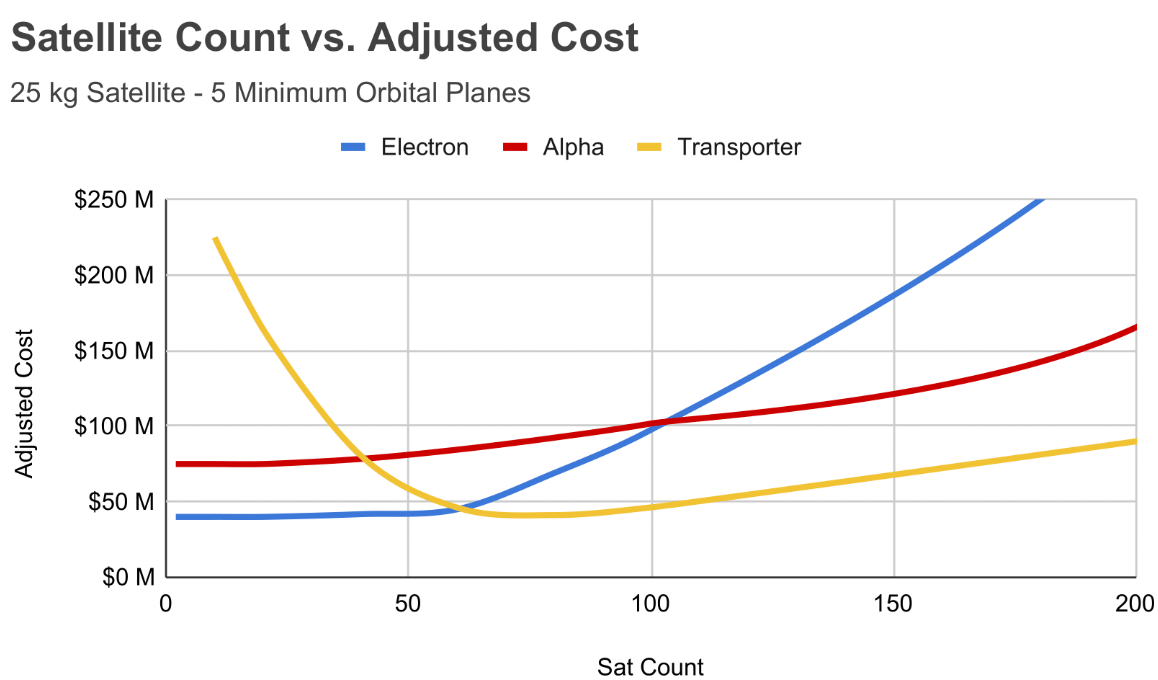
This is essential data to understand the implications of the data below.

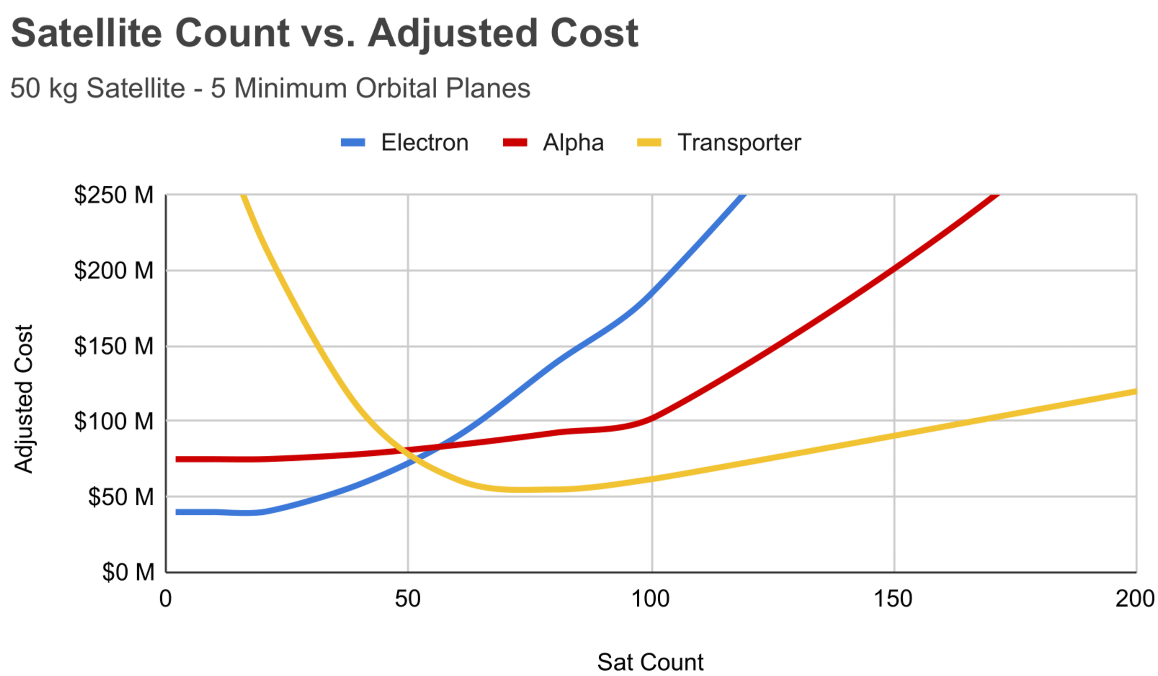
There are three primary categories of small satellite constellations.

1. Constellations with >100 satellites, most satellites under 5kg (eg. PlanetLabs)
2. 10-50 satellite constellations, mass between 30-112kg (eg. Capella)
3. <10 satellite constellation, <15kg mass (eg. Tropics)

The most interesting data I've gathered from this exercise applies to the second category. These are constellations that fit into the category of either 25 or 50 satellites total with masses of either 50kg or 100kg. This is the biggest category of the market available to small sat launch providers.

2. The Tradeoff Between Electron and Rideshare Occurs at ~50 Satellites

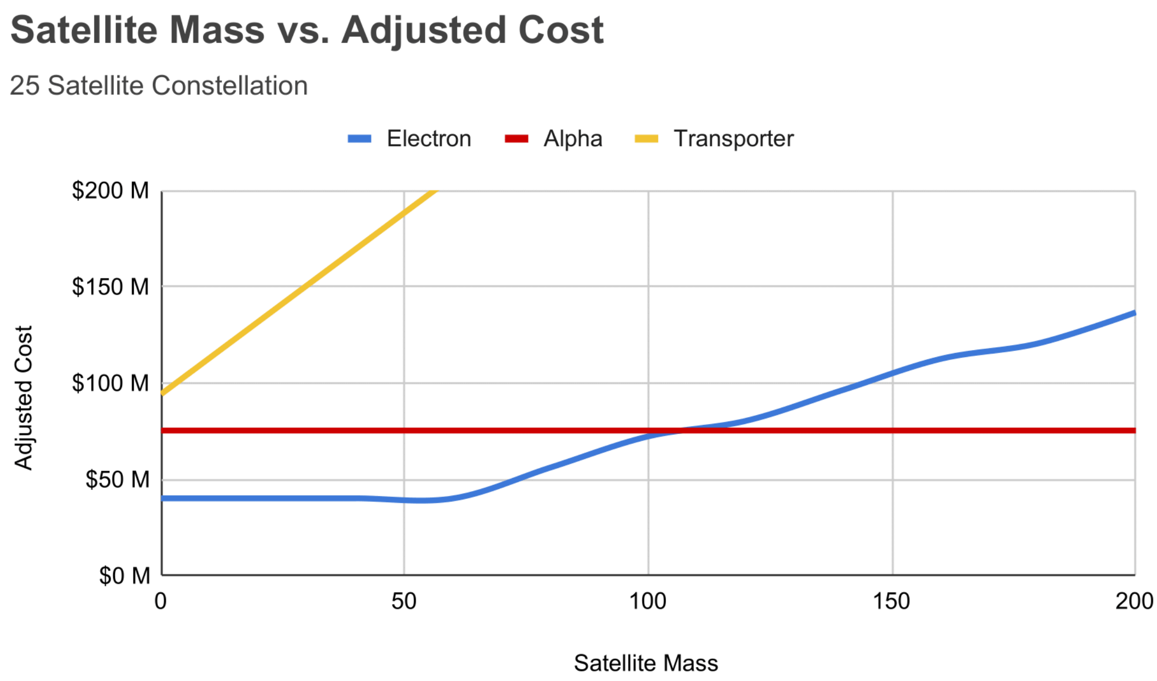




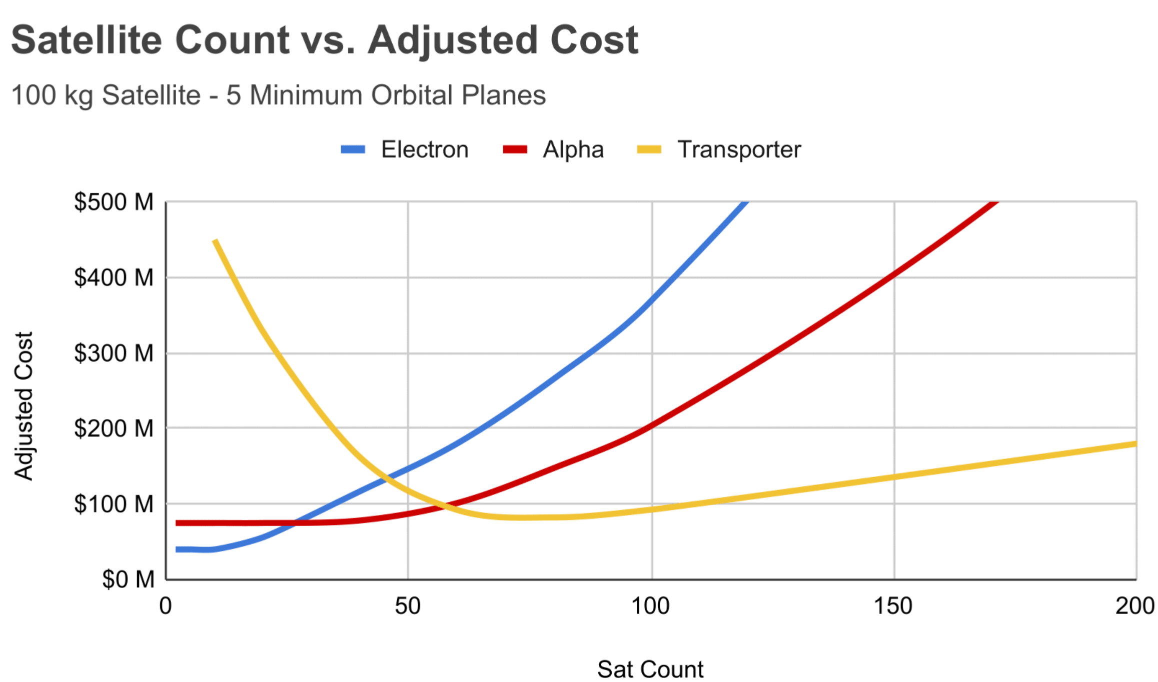
For constellations that don't require the higher payload capacity of a 1-ton class launch vehicle like Firefly's Alpha or ABL's RS1, the options available are either Electron or Falcon 9 Transporter missions. These <50kg satellites are the most common size of constellations and the tradeoff between Electron and Rideshare occurs at ~50 satellites. There are no constellations at this 50 satellite size in my dataset because this is one of the gaps in the market. Go bigger and you end up with Planet Labs, go smaller and you end up with BlackSky at ~20 satellites. This conclusion is present in both the data of real constellations above and in the charts I've created.

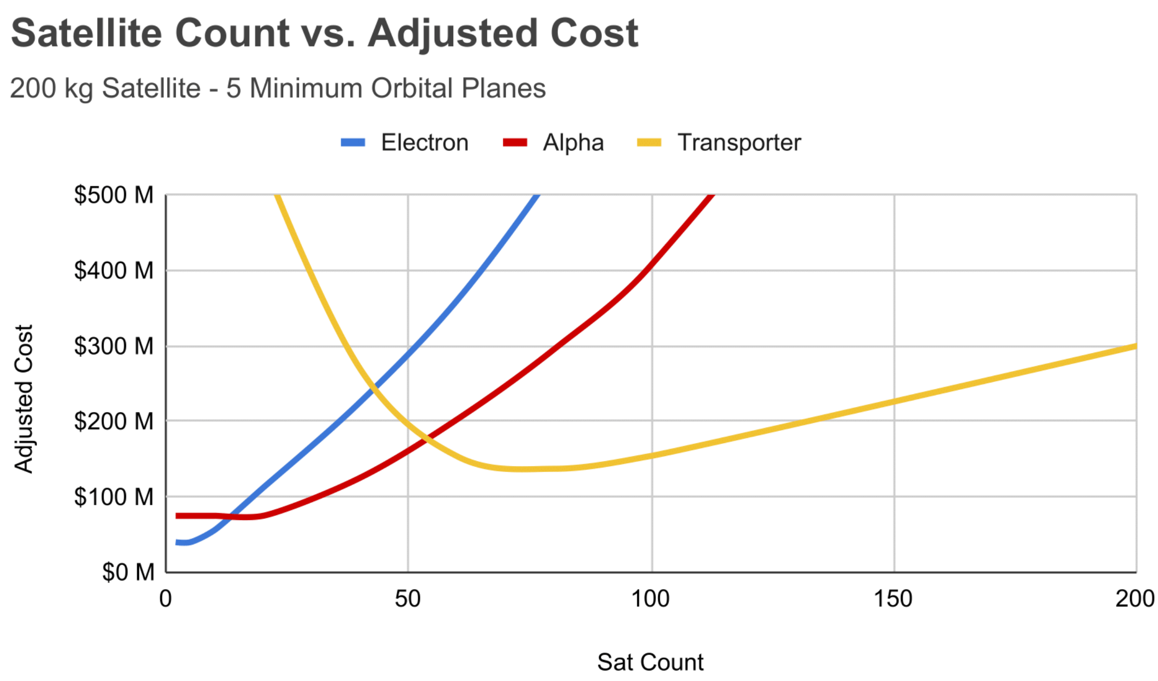
It's remarkable there's enough satellite constellations that I can make statements like the one above. I was born at the perfect time to bask in the glory of the growth of commercial spaceflight.

3. At The Most Common Constellation Size, Alpha Is Optimal for 100kg+ satellites



For most constellations (~25 satellites), Alpha is unable to properly compete with Electron because the satellites are not heavy enough to take advantage of the 1-ton payload capacity of Alpha. The higher payload capacity only starts to kick in with satellites that are over 100kg. For even heavier satellites (eg. 200kg+), Alpha provides a cheaper path to orbit than Electron, we may see some constellations pop up in this category in the future if Firefly or ABL demonstrate reliability. This is only true if the higher satellite mass prodives a significant enough advantage. For example, Earth Observation satellites seem to level out at around 100kg.





For larger constellations - eg. 50 satellites - the tradeoff shifts to lower mass satellites. This benefit extends until rideshare takes over at very large constellations, Eg. Swarm or Spire Global. The reason for this is that with a low number of orbital planes (I used 5 as a default value, we can debate this) a single 1-ton launch is more efficient than several 300kg Electron launches. Firefly's Alpha has ~3x the payload capacity of Electron for ~2x the price. In short, larger constellations benefit Firefly (until they don't) and more orbital planes benefit Rocket Lab.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Small Sat Constellations: The line between Electron and Rideshare

Jul 4, 2024 • Christopher Kalitin

Data ends Dec 31 2023 (mostly)

Constellation / Company	Category	Sat Count	Launches	Sat Mass	Launch Vehicle	Orbits	Rideshare Addressable	Unique Plane Required	Planned Sats	Notes
Planet Labs	Observation	542	29	5.0 kg	Soyuz, Antares, I	LEO / Rideshare SSO	Yes (Scale)	No (Scale)	543+	
Spire Global	Communications	166	34	4.0 kg	PSLV, Atlas V, A	LEO	Yes (Scale)	No (Scale)	166+	Many uses: Ship tracking, m
Swarm	Communications	150	13	0.3 kg	Vega, Electron, F	Unique Plane / LEO / S	Yes (Scale)	No (Scale)	320*	Acquired by SpaceX
ICEYE	SAR	31	12	85.0 kg	Soyuz, Falcon 9	Rideshare SSO	Yes	No (Intermittent Data)	31+	Entirely F9
HawkEye 360	RF Situational Awar	21	7	30.0 kg	Falcon 9, Electro	Rideshare SSO / Bandw	Yes	No (Intermittent Data)	~33	Lately only F9
Black Sky	Observation	19	12	56.0 kg	PSLV, Falcon 9, I	LEO	No (Lack of Scale)	Yes (Lack of Scale)	60*	Lately only Electron, old sour
Capella	SAR	11	11	112.0 kg	Electron, Falcon	LEO / Rideshare SSO	No (Lack of Scale)	Yes (Lack of Scale)	30	
Fleet Space Centauri	Communications	6	6	35.0 kg	Falcon 9, PSLV, I	Rideshare SSO	Yes (Scale)	No (Scale)	140	Mainly Tech Demos so far
Kineis	Communications	5	1	30.0 kg	Electron	High SSO	No (Lack of Scale)	Yes (Lack of Scale)	25	
Tropics	Observation	4	2	5.3 kg	Electron	Unique Plane	No	Yes	N/A	
iQPS	SAR	4	4	100.0 kg	Epsilon, Electron	Rideshare SSO	No (Lack of Scale)	Yes (Lack of Scale)	36	Mainly Tech Demos so far
Synspective Strix	SAR	3	3	100.0 kg	Electron	Unique Plane SSO	No (Lack of Scale)	Yes (Lack of Scale)	25	
Prefire	Observation	2	2	15.0 kg	Electron	Unique Plane	No	Yes	N/A	

Expanded Chart - Spreadsheet

If you have any feedback or criticism, please reply here.

Which Constellations Require Dedicated Launches?

Constellations that have a relatively small number of satellites and require high & regular revisit rates are well-suited for dedicated launches. Many of the recent Synthetic Aperture Radar constellations fall into this category such as Capella, iQPS, Synspective, and Black Sky (Earth Obv). These constellations benefit from high & regular revisit rates so they can provide near real-time data to their customers and cover specific parts of the Earth regularly (eg. once every half hour).

Recently there's been a great example of the Commercial benefit that dedicated launches provide: Synspective signed a 10-launch deal with Rocket Lab. Given an Electron average sales price of $8M, this is a $80M contract. The cost of a SpaceX rideshare mission is an initial charge of $300k for 50kg to SSO with each additional kilogram at $6k. Assuming Synspective launches 2 100kg Strix satellites per Electron this is 20 satellites in total. On SpaceX rideshare missions this would cost $12M, a $68M cost delta.

Launch schedule is another factor to consider when analysing Synspective's choice to go with Rocket Lab. The Electron launches will occur between 2025 and 2027. SpaceX rideshare missions are rumoured to be booked 2 years out and they launch 3-5 times per year (Up to 5 with

Bandwagon rideshare missions). Assuming a start date in 2026, Completing 10 launches for Synspective could take SpaceX until 2028 or 2029. However, this time can be shortened if the satellites have sufficient on-board propulsion to change their orbital planes (This is covered in the final section). Even assuming the schedule issue is resolved with a higher scale in the future, rideshare missions still mean suboptimal orbits.

The cost advantage of rideshare missions is cancelled out by the schedule (temporary) and orbit/delivery (permanent) disadvantages. One of the primary disadvantages of rideshare missions is having to negotiate with several other companies on which orbit the satellites will be deployed into. This means every customer is not where they want to be and on top of the schedule disadvantage, this means each satellite is initially producing less revenue than it otherwise could be. In the case of Synspective, the advantage of launching on Electron appears to be worth at least $3.4M.

Prefire and Tropics do not fall into the category laid out above because they are made of 2 & 4 satellites respectively and will not be expanded in the future. Unlike commercial constellations, these are one-off NASA constellations that serve a very narrow and specific Earth Observation purpose. So, there will not be an increase in demand for the data these satellites provide. When new data is needed, NASA will develop and launch a new set of 2-10 satellites.

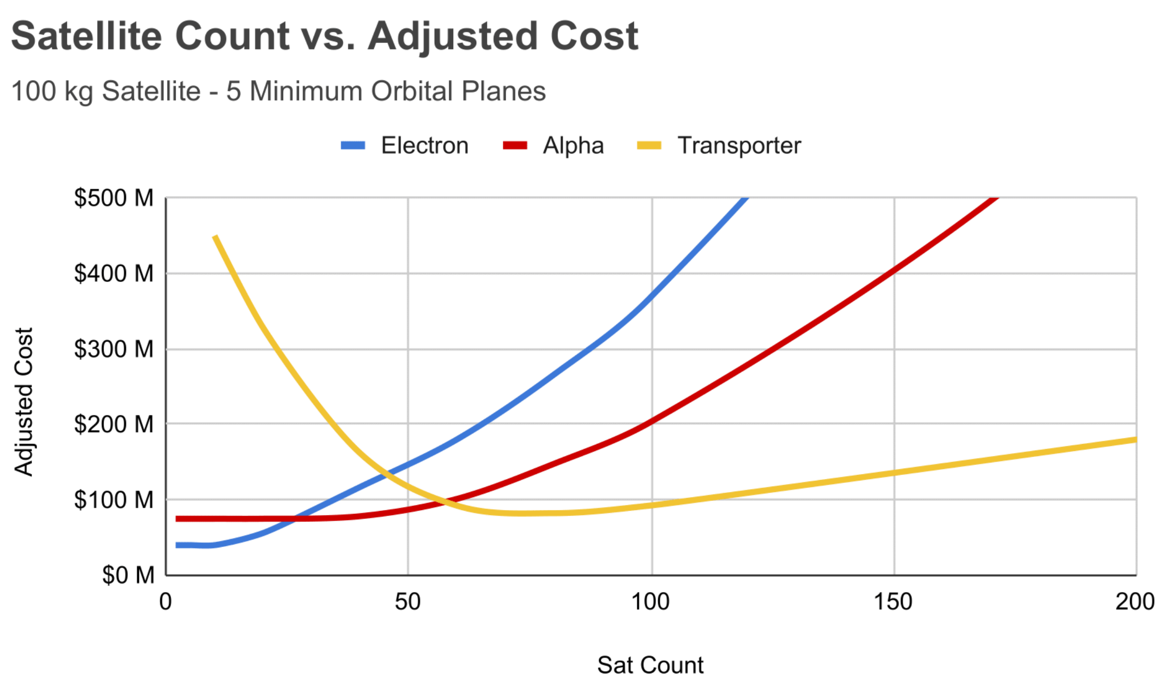
Rideshare Missions Launch Most Constellations Satellites

The first category of Small Satellite Constellations that launch on rideshare missions are those with such high scale that the exact orbits of dedicated launches are not required. These include Planet Labs, Spire Global, and Swarm (before SpaceX acquired them). Planet Labs has 542 satellites, Spire has 166, and Swarm had 150 before acquisition. Given the number of launches required to deploy these constellations, you get a large enough distribution of orbital planes that you can cover all of the Earth with a high revisit rate. Given the falling price of launch, creating satellites cheap enough to produce hundreds so you can utilize rideshare missions could be a winning strategy (Like Planet Labs).

The second category is constellations that don't require the revisit rate benefit of dedicated launches (even spacing of orbital planes). HawkEye 360 is the best example of this as this is one of the few constellations that is not focused on Earth Observation or communication with Earth-based assets. They sell Radio Frequency signal location data. This process involves detecting RF emissions and triangulating the source with multiple satellite passes over the same location. This means the orbit requirements are not as strict as other constellations. In contrast to the Synspective example in the previous section, HawkEye 360 does not get as much direct monetary benefit from dedicated launches. They recently cancelled a few of their upcoming Electron launches.

One of the great benefits of rideshare missions (even to Electron) has been the ability to launch test satellites for a low cost. Many early Capella and BlackSky satellites launched on Falcon 9 and this allowed for cheaper development of their constellations which ended up launching on Electron. This allows for cheaper launches during the early scaling of the constellation and an accelerated completion timeline.

The Market for 1-Ton Class Launch Vehicles



Just like Falcon 9, Electron owns the small sat market because there are no capable competitors. Over the next few years, we'll see Firefly ramp up its Alpha launches and ABL's RS1 come online (hopefully). From a product perspective, these 1-ton class rockets have the potential to take market share in the small-sat constellation launch market. Ozan Bellik pointed this out when I mentioned my blog post Comparing Demand for Firefly's Alpha vs. Electron. My conclusion wasn't completely incorrect, but inaccurate enough to warrant this section.

The niche of the market that these vehicles can solve is constellations with ~100kg satellites that plan to launch more than 20-30 satellites. These include iQPS, Synspective, Capella, and BlackSky. Alpha and RS1 cost around $15M which is ~2x more than Electron while having ~4x the payload capacity. On the surface, this appears to be a 2x improvement in $/kg, but the entire capacity of the rocket may not be used. Using the entire 1000kg of payload capacity would mean launching ~10 satellites at a time which negates the advantage of having satellites spaced out in different orbital planes which lowers revisit rates which in turn makes a constellations offering less competitive with other Earth Observation companies, ie. lower revenue. The breakeven point for a 1-ton class rocket vs. Electron occurs when 4-6 100kg satellites are launched on a single mission, 2x the payload for 2x the cost.

Rocket Lab has the massive advantage of high cadence and a proven track record, so in reality, the breakeven point from a constellation operator's perspective goes even higher. The massive first-mover advantage Rocket Lab has today translates to other aspects as well. In Peter Beck's recent interview with Eric Berger in reference to constellation customers he said: "They've designed their constellation or their spacecraft around Electron. It does things that you just can't get on other missions." Given the size of recent constellations in both individual satellite mass and the total number of satellites, it's clear that Electron is the best option and is being optimized for by the industry. It will take many years for the advantages of 1-ton class rockets to be fully realized by the industry.

iQPS's upcoming constellation may be the only one out of the 7 dedicated-launch-addressable constellations in the chart that could reasonably fly on a 1-ton class launch vehicle. According to Gunter's Space Page (Amazing source) they are planning a constellation of 36 100kg satellites. With 6 launches on a $15M 1-ton class rocket, this could be competitive with Electron. The other constellations either have satellites that are too light to warrant anything other than Electron or too few planned satellites. Too few satellites mean unoptimized coverage of the Earth if launching too many at once, ie. lower revisit rates.

Given the lack of constellations launching in the next few years that would see a large enough benefit from 1-ton class rockets, it's unlikely Alpha and RS1 will reap the benefits of their size & cost per kilogram advantages until the late 2020s. Assuming constellation design shifts to the 1-ton class instead of Electron due to cost, there is a world in which they get 50% or more of the market. However, at a certain size of constellation rideshare missions become competitive.

The Future of Launching Small Sat Constellations

There is zero chance Electron loses a significant number of launches to another rocket in the next ~3 years. As Peter Beck said, they are the industry standard in small satellite launch and the industry is building around the capability Electron provides. However, in 5+ years if the 1-ton class rockets succeed, there will be more room in the industry to design larger satellites that take advantage of the cost per kilogram advantage of these rockets. The lower cost of mass also allows for heavier (cheaper) materials to be used, lowering satellite costs even further. This is however not a paradigm shift as it is mainly a marginal improvement. There is enough space for Firefly or ABL to succeed in the market, but it's very unlikely they displace Electron. The market will grow to support more companies, market share may be lost, but there will be net growth. Rocket Lab enabled the small sat business while Firefly and ABL are merely improving it.

Rideshare missions will soon have the ability to eat even more of the dedicated small sat constellation launch market when kick stages come online and propulsion becomes cheaper. The two main advantages of dedicated launches are schedule and orbit precision, all other benefits stem from these points. Schedule will remain a benefit of dedicated launch but orbit precision can be addressed by kick stages or on-satellite propulsion.

Sun Synchronous Orbits precess around the Earth which is the mechanism that allows them to always stay in the sun, otherwise, there would only be a few days per year in which the polar orbit is 100% exposed to sunlight. This same principle makes it possible to shift the orbital plane of a satellite with minimal fuel expenditure through slightly increasing altitude and changing the inclination of the orbit. This process can take up to several months or if the launch is already near the desired orbital plane a few weeks is possible. Either way, revenue is delayed. However, for mature constellations, revenue delay is less of a concern and millions of dollars per satellite can potentially be saved.

Either space tugs like Impulse's Mira or ion thrusters on each satellite can accomplish the low-energy polar orbital plane change. Ion thrusters have the advantage of providing efficient propulsion that can extend the satellite's lifespan. Starlink satellites currently use Argon ion thrusters to boost themselves to their final orbit. With 2500 seconds of isp, they are far more efficient than chemical propulsion, meaning more mass can be delivered to the final orbit. As always, the limiting factor here is the cost. This is possibly offset by a longer satellite lifespan, but it remains to be seen how many constellations opt for this route. In the next 5-10 years we will see.

The advantage of ion propulsion is best seen on large constellations like Starlink, OneWeb, or Kuiper. This may have an impact on the industry as we see convergence to an end-to-end model. In the case of Rocket Lab, this could mean manufacturing their Lightning satellite bus with ion propulsion to host many different payloads. This allows for the benefits Starlink has seen with a common satellite bus and launch vehicle while serving multiple customers. While Rocket Lab sees competition from 1-ton class rockets and potentially Stoke's Nova, they will be transitioning to hosting payloads on a common satellite bus where Neutron and Space Systems take over from Electron on constellations.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

On the Overallocation of Labour to Finance and Law

May 25, 2024 • Christopher Kalitin

Go to where the vibes are

Useless ambitious people are the most dangerous people in the world

The value of finance and law professionals is found in the facilitation of real useful work. The aim of increasing the efficiency of a productive society is to lower the percentage of accountants and lawyers to the minimum required amount. Increasing the number of these useless workers and increasing their power leads to negative outcomes. Many issues in modern society are attributable to this class in two ways: (1) inefficiency leads to suffering and (2) jobs that don't inherently produce value are susceptible to becoming part of the Professional Managerial Class.

Hospitals demonstrate the suffering of inefficiency. There are countless reasons government-sponsored institutions are inefficient, but in the case of hospitals finance people are doubly to blame through insurance. Increasing efficiency is the goal, **do not be complicit in these systems.**

The Professional Managerial Class is a part of the the deep state. A decentralized system of middle men and beauracrats that are not inherently useful themselves. Decentralization is a point of the deep state that must be highlighted (I am using the word deep state to mean groups or individuals with power over society that aren't obvious). Every group has ways of profiting off of events and movements around the world and this sets the incentive structure. Powerful groups (politicians, corporations, PMCs (both meanings)) are incentivized to support certain outcomes of movements. They piggy back on these movements to support their own goals - not quite hijacking but slightly nudging in their preferred direction. This means the teachers making your kids gay are not doing it because of a secret meeting they had with Big Gay. They are doing it because they think it is the right thing to do and this is supported by deep state groups, sometimes quite obviously.

How many watts do you use?

We must to chose which metric to work on increasing to expand society. The economy or the S&P 500 aren't the correct metric because it is very scammable as we've seen in the last 5 years. On the other end of the scammable spectrum is total energy output and to a lesser extent gross domestic product.

The S&P 500 can't be the metric we use because it's scammable by the government. We must avoid scammable systems because they incentivize scammers. Lawyers convince you they are needed by supporting the creation of laws. A system of arbitrating disagreements is needed, but remember, efficiency is the goal. The Professional Managerial Class are scammers in that they do not provide value themselves but rather hijack systems that provide useful output. Energy output measures what we truly produce and is one of the least scammable metrics. This is why the Kardashev Scale is the ultimate measure of a society.

Increasing output increases human flourishing

I was careful when writing the title of this section to not say: "Increasing output increases human happiness."

However, increasing productive output does increasing the flourishing of humanity. We can explore the universe and connect people on opposite sides of the planet. Increasing the tech level of humanity leads to a higher potential for happiness, exploration, and everything else. One of the core outputs of humanity has been increasing the time and impact of leisure (Think heating, soft beds, TVs, TikTok, Starbucks). This will hopefully increase happiness.

If everyone followed Andrew Tate's teachings you would have no engineers. Tate is part of the group that harnesses technology to increase either their own economic worth and/or power. Someone has to harness technology and there are many more individuals that play this role rather than those who build technology. Building technology is more useful - and harder - than harnessing it.

Don't be a scammer

A scammer is an individual who does not provide value themselves but rather forcefully extracts value from productive systems. Think back to finance bros and lawyers.

One of the best outcomes of capitalism is giving the option to harness selfishness for the good of society. However, this only occurs when this fundamental rule of capitalism is followed: goods, services, and currency are exchanged *consensually*. Increasing the efficiency of society means decreasing the number of scammers - those who do not follow the above rule of capitalism, exchanges are consensual.

Every scammer is an individual that could be increasing total output instead of manipulating the systems built around real productivity to favour themselves. Atoms cannot be scammed. The closer you get to the base level of reality the less scamming is possible and the more impactful the work is.

Misallocating the time of scammers to scamming - which depending on their field requires immense intelligence - is a waste of human capital. Building the core productive systems that scammers exploit is far more useful than being one of the individuals that extracts value from these systems for their own benefit, you also get more rich. The inefficiency of scammers currently infects and will continue infect AI. Many H100s are wasted on activities that are not increasing productivity.

BUILD

The four principles laid out above:

1. Do not be complicit in increasing the inefficiency of the world
2. Do not create or work on scammable systems and minimize your participation in them
3. Increasing productive output increases human flourishing
4. Instead of scamming, BUILD!

Increasing the productive output of humanity by building useful things is one of the best uses of life. Your creations will directly impact the future flourishing of humanity, or, if you work on something farther from the base level of reality, you will at least be one of the real productive people - someone has to do the real work. A life well lived is one directly useful to the future of humanity.

Deep tech is the least scammable sector and the place where the most true impact and innovation occurs. This is why You Should Be Working On Hardware.

Subscribe

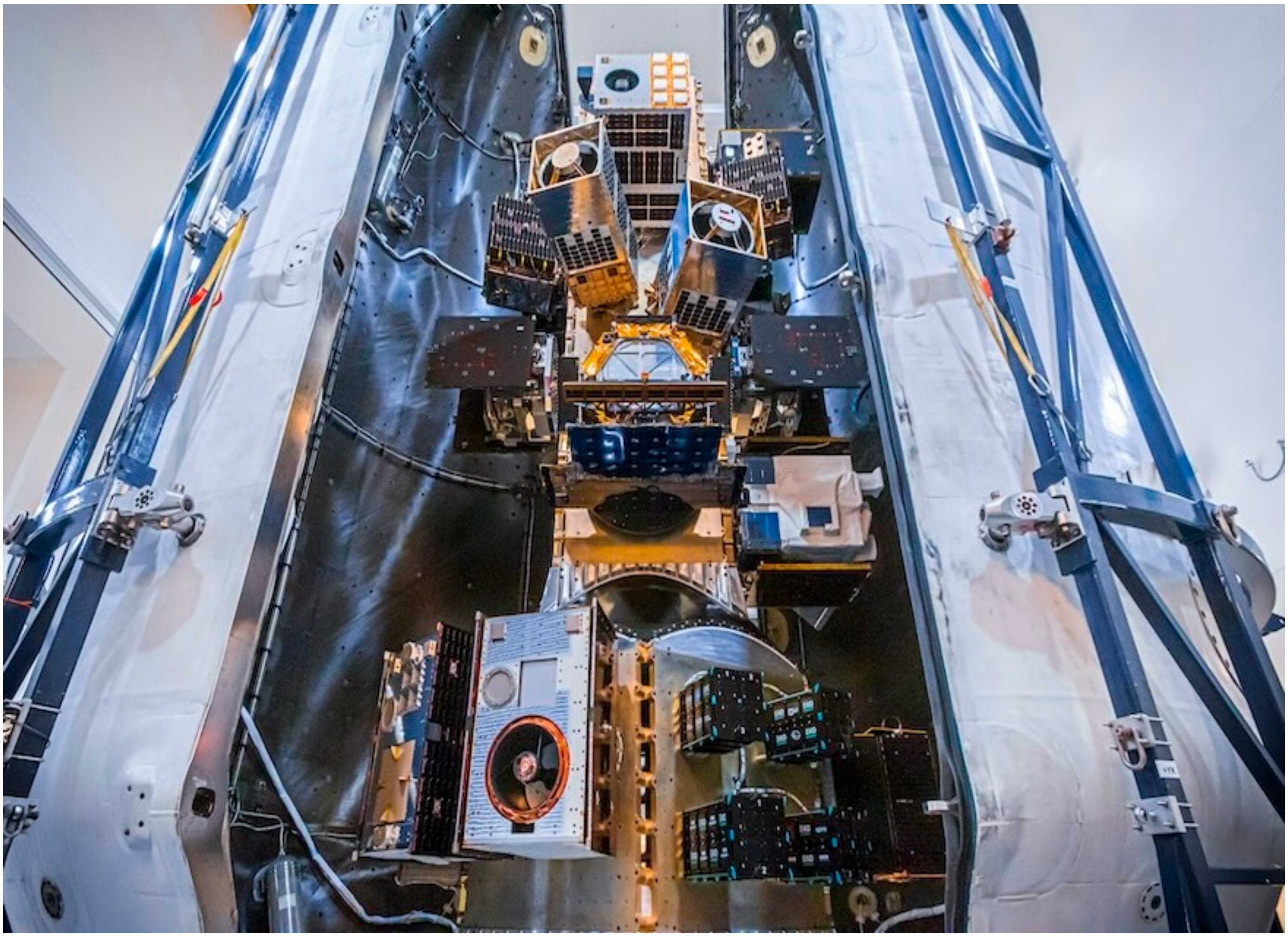
CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

What SpaceX's Bandwagon Missions Mean for Small Rockets

Apr 9, 2024 • Christopher Kalitin



SpaceX's Bandwagon-1 rideshare mission launched yesterday and I was busy speedrunning a unit of Chemistry 11 in 8 hours to write this blog post before the launch. Regardless, this is the first of SpaceX's rideshare missions to low inclination orbits. The Bandwagon-1 mission will expand the range of payloads that can be launched on a Falcon 9 rideshare mission and will continue the trend of making dedicated small satellite launch a smaller niche of the market. However, hile dedicated small satellite launch will lose market share, the overall market will increase and provide more satellites that must be launched.

Three months ago I wrote this blog post on the dedicated small sat launch market. The Bandwagon missions will have significant implications on the conclusions I came to. Specifically, many of the satellites I listed as requiring a dedicated small sat launch in fact do not.

Payloads Unsuited for Dedicated Launch

Bandwagon-1 launched 11 satellites, from SpaceX’s website: “On board this mission are 11 spacecraft including KOREA’s 425Sat, HawkEye 360’s Clusters 8 & 9, Tyvak International’s CENTAURI-6, iQPS’s QPS-SAR-7 TSUKUYOMI-II, Capella Space’s Capella-14, and Tata Advanced Systems Limited’s TSAT-1A.”

Two of the payloads are very well-suited to a rideshare mission:

1. CENTAURI-6 - Internet of Things connectivity satellite, 12kg
2. TSAT-1A - Tech demo for Earth Observation, 42kg

The CENTAURI-6 satellite is a 35kg, 6U cubesat that provides connectivity for relatively low-bandwidth applications. The maker of the satellite, Fleet Space Technologies, plans on launching a constellation of 140 satellites. Previous prototypes have flown on a PSLV, Electron rideshare, and three times on Falcon 9 rideshare missions. Because these are prototypes, they do not require exact orbits like a full constellation would. SpaceX Transporter missions are ~10x cheaper per kg than a dedicated launch, so it makes sense to launch these prototypes on a rideshare mission. In the future, when the constellation is ready to be deployed, they will likely use dedicated launches to get full coverage of the Earth.

The TSAT-1A satellite weighs 42kg and is a tech demo for Earth Observation from India. This is a demonstration satellite and does not require exact orbits. Given the fast that rideshare is 10x cheaper than a dedicated launch, it’s clear why these didn’t launch on an electron.

The default for small satellites is to use a rideshare mission, they only use dedicated launches if it is required for their mission or if they have schedule constraints.

Payloads That Could’ve Launched on a Small Launch Vehicle

1. Korea 425 - Korean Military SAR satellite, 800kg
2. HawkEye 360 Satellites - Space Situational Awareness Satellites, 25kg each
3. QPS-SAR-7 TSUKUYOMI-II - SAR satellite, 120kg
4. Cappela 14 - Earth Observation Satellite, ~100kg

The Korea 425 Satellite is the second in a line of Synthetic Aperature Radar (SAR) satellites for the Korean military. It’s mass is 800kg which is outside the range of many small launch vehicles like

Electron, but Firefly's Alpha or ABL's RS1 could have launched it assuming the payload fairing are large enough. In the future, if the Korean military wants to position their satellites to optimize coverage of the Earth, they may need to launch on a dedicated mission.

Last year, on December 1st 2023, the first Korea 425 satellite was launched on a Falcon 9 mission along with 23 small satellites as rideshare payloads. Because it was the primary payload on the mission it was able to dictate the orbit and inclination of the mission. This may have occured on Bandwagon-1 as well because the Korea 425 satellite is the most massive payload on the mission and the first one listed by SpaceX.

When 1-ton class launch vehicles start launching regularly, it may be more cost-effective to launch these 1-ton class satellites on those vehicles. However, this time the Korean military chose to launch on a Falcon 9 rideshare mission likely because RS1 and Alpha were not an option at the time.

Six Hawk satellites were launched on this mission. In my previous blog post on the small satellite launch market I categorized these as Earth Observation when in fact they are Space Situational Awareness satellites. They collect RF signal data in space and sell this data to commercial customers. Previously they have flown on an Electron and on six Transporter missions. These satellites have been mostly indifferent to inclination and their exact orbit, but as they scale the constellation they may perfer dedicated launches to get better coverage of LEO. They have 2 launches coming up on Electron.

The trade-off between frequent coverage of the Earth and cost is clear in the case of iQPS. Electron has launched many Earth Observation satellites. This is their primary market because many constellations seek to image parts of the Earth as frequently as possible. SpaceX calls this a "high revisit rate". The QPS-SAR-7 TSUKUYOMI-II satellite is a Synthetic Aperature Radar satellite and iQPS has previously launched these satellites on a dedicated Electron mission as well as a rideshare missions on Epsilon and Falcon 9. Because of the relatively small size of the constellation - they're on their 7th satellite - they don't need dedicated launches to optimize their coverage of the Earth as opposed to a constellation like Starlink that must get very high coverage.

Capella has previously launched satellites on both Electron and Falcon 9. They currently have 7 satellites in orbit so they fall into the category of constellations small enough that they don't require dedicated launches to achieve optimal coverage of the Earth. The reason for Capella launching on both Electron and Falcon 9 is likely schedule. They may want to reasonably space out their satellites - hence launching one at a time - but also want to launch many. So, a few Transporter missions a year is not enough.

The advantage of Bandwagon missions over Transporter can be seen when comparing the inclinations of Capella's dedicated Electron missions vs. the Falcon 9 missions. The Capella Electron launches had an inclination of ~45 degrees while Transporter is around 95 degrees. The lower inclination means the satellites spend less time at the poles and more time over economially interesting areas of the Earth.

Analysing Electron's Manifest

There are a couple sources for Electron's 2024 manifest. Many of these are outdated as the manifest has changed since they were published. The most up-to-date source I've found is Scott O's list, even better than Next Space Flight's.

Live and Let Fly	March 21, 2024	National Reconnaissance Office
NASA / KAIST Rideshare	April 24, 2024	NASA / KAIST
HawkEye-2	June 2024	HawkEye 360
Kinéis-1	June 2024	Kinéis
BlackSky-07	July 2024	BlackSky
NorthStar-2	H2 2024	Spire & NorthStar
PREFIRE-1	August 2024	NASA
PREFIRE-2	August 2024	NASA
Kinéis-2	September 2024	Kinéis
Kinéis-3	November 2024	Kinéis
Winnebago-2	Q3 2024	Varda Space

In the previous secitons I laid out the advantages of Transporter / Bandwagon missions. Both are lower cost and Bandwagon gives a more ideal orbit for many payloads because of the higher revisit rates. Now these advantages can be used as a framework for analysing which types of payloads on Electrons manifest may be at risk. Not the payloads themselves, but the general types of payloads.

The first launch on the list, NROL-199 (NROL-123?) "Live and Let Fly", was a US military launch. The military uses dedicated launch because of security reasons with payloads and this payload may need an exact orbit. From Jonathan McDowell's data it's not clear the final orbit the satellite was deployed into (COSPAR ID: 2024-053A). The estimated orbit is 500km at 48 degrees.

The NASA / KAIST (Korea Advanced Institute of Science and Technology) rideshare mission is going to a specific orbit. First, KAIST's NEONSAT-1 will be deployed in a 500km orbit and then the Photon Kick Stage will be used to deploy ACS3 - a solar sail tech demo satellite - to a 1000km orbit. From the mission profile, it seems the ACS3 satellite is the primary payload. The KAIST satellite may be launching as a rideshare payloads on an Electron due to schedule constraints, while the ACS3 satellite requires a dedicated launch for a specific orbit. Kick stages like Helios's Mira will made it possible to launch these payloads on Medium-lift rideshare missions like Transporter. In the cost estimates section of my previous blog post I point out how this can be much cheaper than a dedicated launch.

NASA's PREFIRE launches are an excellent example of the perfect payloads for Electron. The PREFIRE-1 and PREFIRE-2 satellites are Earth Observation satellites that will be deployed to a polar orbit to study the Earth's poles. Like NASA's earlier TROPICS launches, these satellites require specific orbital planes to optimally cover the Earth. This can't be done on a rideshare mission unless you are the primary payload, like the first launch of Korea 425. This is the type of launch that will continue to be done on dedicated small launch vehicles.

The remaining payloads on Electron's manifest can be put into two categories. First, payloads that are part of relatively small constellations that don't require perfect seperation of orbital planes. Second, payloads that much prefer unique orbital planes which can't be achieved on rideshare missions.

Both the HawkEye 360 and BlackSky have launched on both rideshare and dedicated launches in the past. Because of the significant cost savings over dedicated launches, they may choose to launch on Transporter or Bandwagon missions in the future, as they have in the past. Particularly Bandwagon in the cases of these two companies as they both benefit from lower inclinations. However, as they scale their constellations the benefit of unique orbital planes for each group of constellation satellites starts to outweigh the cost savings of rideshare.

Kineis and NorthStar are planning to start with large constellations. They aren't quite following the exact same path of previous satellite constellation companies like Capella which began with many test satellites. Right off the bat they've booked 8 Electron launches between them and may continue to book more dedicated launches as they scale their constellations. This approach allows a faster ramp to profitability. For some, this approach was not as possible in the past because of the youth of the small launch market. For example, when Capella was launching their first few satellites Electron only had a couple of launches and the industry had much less experience in manufacturing reliable small satellites. Like PREFIRE, these small satellite consetllations are the type of payloads that are very well-suited to dedicated small launch vehicles.

Just yesterday Rocket Lab signed a contract to launch a US Space Force payload to a very low Earth Orbit. "DiskSat will demonstrate sustained VLEO (very low earth orbit) flight and test a unique, 40-inch diameter, disk-shaped satellite bus that is designed to increase on-orbit persistence," the Space Systems Command said. This is a unique payload that requires a dedicated launch and it's where Electron shines. In the future this may be addressable by a kick stage like Mira on a rideshare mission, but for now it requires a dedicated launch.

Outlook on the Small Satellite Launch Market

In my Analysing the Dedicated Small Sat Launch Market blog post I came to a slightly incorrect conclusion. This error came from looking too closely at Rocket Lab and not the market as a whole as I did not look closely enough at the rideshare market to see how many small constellation satellites don't need exact orbits. My conclusion was that almost all small satellite constellations would require dedicated launches. This is slightly incorrect as we can see from the HawkEye, BlackSky, etc. launches on rideshare missions. There is an advantage from launching on dedicated missions - as is seen in Kineis and NorthStar's approaches - and with the growth of constellations like BlackSky there is a push for more dedicated launches. I came to the right conclusion, but with incorrect reasoning.

Rideshare inevitably would take significant market share from dedicated launch vehicles. The cost savings are too significant to ignore. Some think that SpaceX is maliciously trying to take market share from Rocket Lab and other small launch providers. This is not the case as rideshare is a commercial inevitability. SpaceX is simply providing a service that is cheaper and more convenient for many small satellite operators.

Rocket Lab does not compete with SpaceX on small satellites. In a conversation with Anthony Colangelo on the MECO podcast (check it out!) several years ago Peter Beck said, "[Dedicated Launch is] never going to be as cheap as rideshare" and "Our vehicle is more expensive and we don't try to compete with rideshare" (47:50).

From looking at Electron manifest we've seen that around 10-20% of payloads will be addressable by rideshare missions in the future. This is offset by the growing small satellite constellations. Rideshare allows for a lower barrier to entry for satellite constellations, which often evolve and expand to needing dedicated launches. There is a significant market for dedicated small launch vehicles even while competition from MLV rideshare grows. Over the next decade, rideshare will not be the largest threat to small launch vehicles. Vehicles like Nova will be the largest threat in my opinion as they will be able to have high cadence because many different segments are available to them which allows lower costs. Just like Falcon 9, it will be a side effect that they destroy the partially reusable small launch market.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

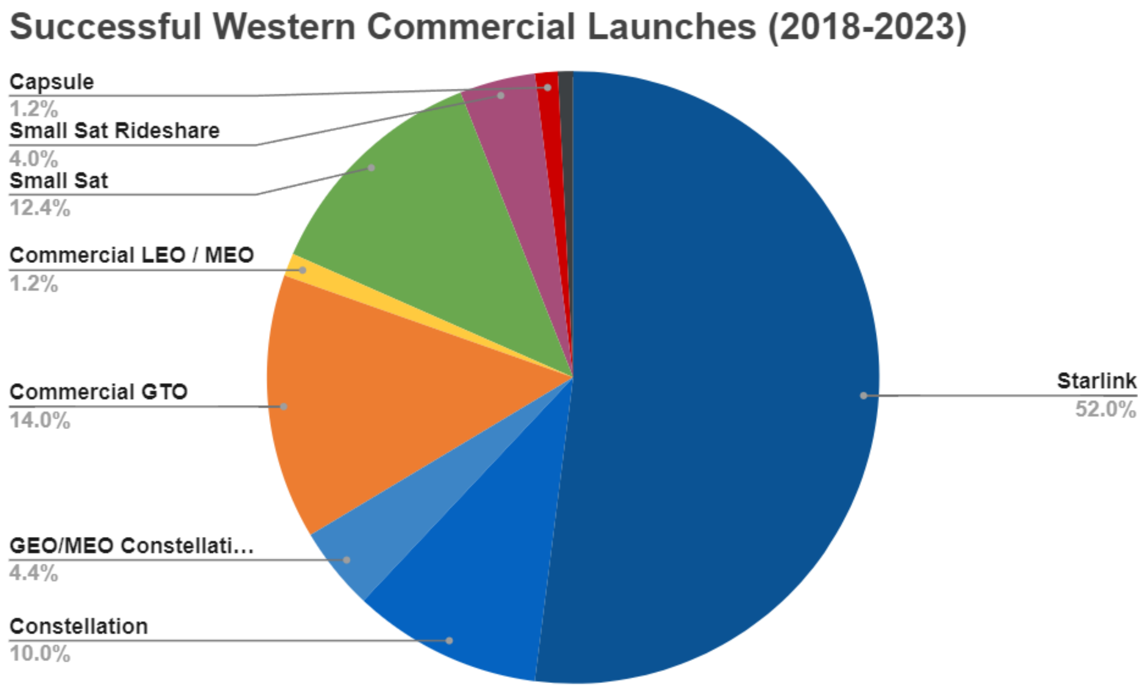
Stoke's Nova is a Perfectly Sized Rocket

Feb 26, 2024 • Christopher Kalitin

At first glance, Stoke’s Nova appears to be an immensely exciting R&D project without much commercial appeal. In a world where the most successful rocket is a 20-ton class partially reusable launch vehicle, it is difficult to see exactly how such a comparatively small rocket will compete. Data is the answer to all your problems, so I’ve answered all my questions with my dataset (Based on Jonathan Mcdowell’s public data, Gunter’s Space Page, and others) on successful western orbital launches from 2018-2023 inclusive and will convey all insights in this blog post. My primary goal is to examine Nova from a demand perspective.

Stoke’s Nova rocket is perfectly sized to compete for existing satellite launches and future constellations. It can compete for LEO satellite launches as the average mass of customer payloads on Falcon 9 is 3.7 tons (ex. Dragon). Full reusability will allow it to compete against Electon and launch constellations even with its 3t-7t LEO capacity. The launch cost must be several times lower than competitors and reusability allows for this. Finally, the fully reusable architecture lends itself to becoming a cargo/crew capsule while refilling in orbit to address high energy orbits and potentially Moon/Mars landings.

Full Reusability Makes Constellations Addressable



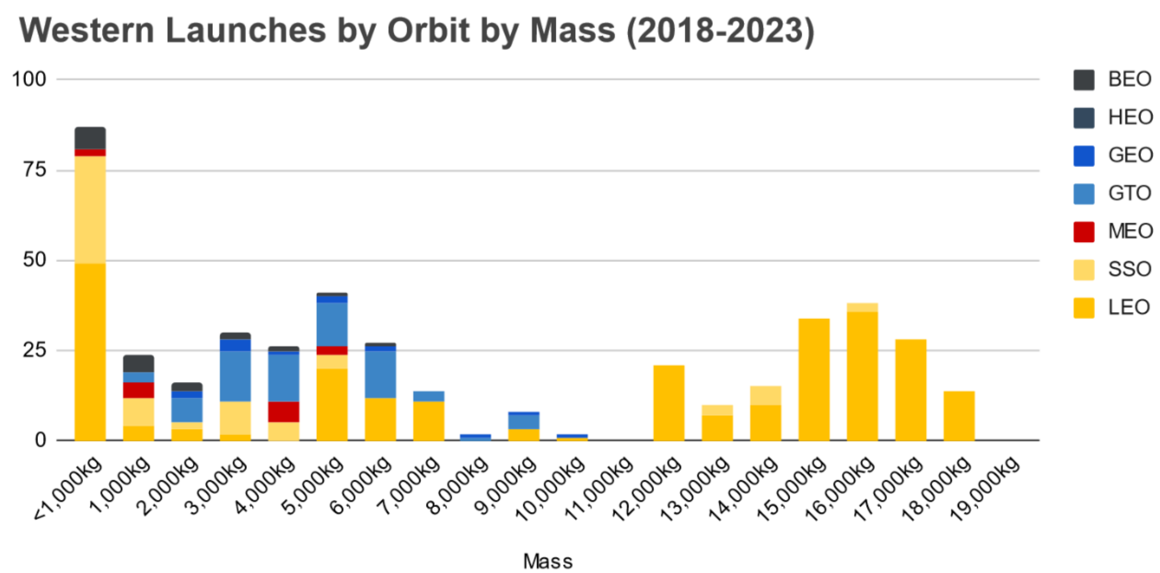
Through following current trends in the launch market it is clear that launching constellations is the market that almost all new rockets should be optimized for. 67% of commercial launches in the last 6 years were for constellations. 80% of these launches were Starlink while the remaining is comprised mainly of Iridium and OneWeb. With numerous upcoming constellations - mainly Kuiper - it is clear this market will experience much more rapid growth than other payload types.

Excluding Starlink, the market for commercial launches is shared in equal thirds between Constellations, GEO Communications Satellites, and Small Satellite launches. Legacy GEO Communications satellites are a fairly static market and have been for ~20 years at around 10 launches per year. Small satellites are experiencing exponential growth, but the TAM for rideshare missions is relatively low. This leaves constellations as the massive market that is exponentially growing.

The primary considerations behind launching constellations are responsiveness and cost per launch per satellite. Constellations can be launched on rockets of almost any size because they are made up of a large number of relatively small satellites. To illustrate this, a 20-ton class medium-lift launch vehicle is not the only vehicle type capable of launching constellations because a 5-ton class rocket can be competitive if the cost is 4x less and has 4x higher cadence. Furthermore, the requirement for an increased number of launches potentially gives advantages in responsiveness.

Stoke’s Nova is rumored to have a 3-ton LEO capacity when reused, 5-tons when the booster is reused, and 7-tons when fully expended. At this scale, to be competitive with Falcon 9 it must be ~6x cheaper per launch which would cost $10M for the customer or ~$2.5M internal marginal launch cost. As we’ve seen with Falcon 9, reusability has dramatically decreased launch costs to the point where the majority of the cost of a rocket is no longer in the first stage. Second-stage reusability will drive down costs even further and a small launch on the scale of Stoke’s Nova is well suited to achieving a high launch cadence which will accelerate the process of decreasing costs.

Nova can Launch Most Payloads



Launching legacy geostationary communications satellites has very different considerations from launching constellations. These launches require immense mass to GTO capability and prefer high-energy optimized architectures like Ariane 5 or Vulcan. However, ~60% of the commercial GEO satellites launched in the last 6 years were under 5 tons. Because refilling in orbit is planned for Nova, it will be able to put its entire LEO capacity directly in a geostationary orbit and potentially return the second stage (Assuming 6000m/s+ delta-v in the upper stage similar to Starship & <3-ton payload). Because the vehicle is fully reusable - at <3-ton payloads - this may cost less than a dedicated Falcon 9 launch.

The average mass of Falcon 9 customer LEO launches excluding Dragon is 3.7 tons. Between 2018 and 2023 inclusive, 32 of these launches occurred and only 4 of them were over 5 tons. This means Nova could have launched 28 of the previous 32 Falcon 9 customer launches while reusing the booster and slightly less if reusing the upper stage as well. This assumes there were no further considerations such as payload volume. Because Falcon 9 is oversized for the majority of LEO customer payloads, Nova is well-positioned to compete in this market.

Small satellite rideshare missions are well suited for Nova given their low mass requirements and flexibility. For example, SpaceX Transporter missions are often below 5 tons, another example of Falcon 9 being oversized for customer payloads. The limiting factor in Nova being able to launch these missions may be payload volume because Nova’s payload fairing is smaller than Falcon 9’s. Regardless, this is solved by launching fewer satellites at once. Rideshare missions are flexible enough that they can launch on nearly any vehicle and the potential for extremely low cost with Nova makes it well-suited for these launches.

It’s not just rideshare missions that Nova can address, but given potentially extremely low costs, it could be economical to conduct dedicated launches for small satellites. Stoke will have to decrease costs below where Electron is today - ~$8M - and increase cadence. If second-stage reuse is achieved, Electron-level costs are possible and because Nova has many more market segments available than Electron, a higher cadence can be achieved.

All 4 of the Falcon 9 LEO customer launches that exceeded Nova’s 5-ton booster-reuse payload capacity were for the US Military. There were two NROL launches and two SDA launches. Because of the classified nature of these payloads, their exact mass is unknown, so some of them may be within Nova’s 5-ton payload capacity. Even so, to launch these payloads Nova would have to be certified by the US Military to launch classified payloads under NSSL Lane 2 which is unlikely given competition from SpaceX and ULA. NSSL Lane 1 payload are of lower mass and lower energy orbits which Nova will be able to compete for.

NSSL’s phase 2 target orbits can illustrate the advantage of orbital refilling. Nova may be able to achieve all but one of the NSSL Phase 2 target orbits if the vehicle used the 7-ton fully expendable LEO capacity. These orbits are mainly high-energy and require 4-7 tons of payload capacity. For example, the GEO 2 target orbit requires 6.6 tons direct to GEO and MEO Direct 1 requires 5.3 tons direct to an 18,200km orbit of Earth. The only required orbit Nova can’t achieve is Polar 2 which requires 17 tons to a 830 km polar orbit of Earth. This shows that in-space refilling is a feature that opens up a lot of capability and how Nova will be competitive for NSSL Phase 3 Lane 1 payloads.

Crew / Cargo Capsule and Lunar Landing

Nova’s upper stage is a fully reusable orbital reentry vehicle capable of propulsive landing. Previous vehicles capable of orbital reentry have all carried crew or cargo and orbital vehicles capable of propulsive landing have all been Moon/Mars landers.

In Stoke Space’s promotional video 4 months ago they teased in-orbit refilling and Moon landings. As mentioned above, in-orbit refilling allows them to increase payload capacity beyond low-earth orbit which also makes it possible to achieve a lunar landing. Using low earth orbit as a starting point, a lunar landing requires ~5,700 m/s of delta-v. If Nova’s upper stage has a similar delta-v to Starship (6000m/s), this is possible with a full 5-ton payload. Even if delta-v is lower than 6000m/s or significant extra mass is required to achieve a lunar landing, the payload capacity to the lunar surface will be greater than a ton. This is a higher payload capacity than the largest Commercial Lunar Payload Services lander - Astrobotic’s Griffin - which is capable of delivering 625kg to the lunar surface. A Nova-based lunar lander could deliver more payload to the Lunar surface for a lower overall cost than the existing CLPS landers.

Lunar landings of lower cost and higher reliability are made possible by testing the landing system on Earth dozens or hundreds of times before attempting a Lunar landing. Engines and maneuvering systems will be tested many times before a lunar landing which makes a failure during landing far less likely. For example, failures like Peregrine, Luna 25, or Beresheet would not have occurred on a sufficiently tested vehicle. Lunar-specific features cannot be tested on Earth such as communications or landing software and these are among the greatest causes for failure. A lunar landing with Nova remains a significant technological challenge, but a vehicle tested on Earth or in Earth orbit hundreds of times has a far higher chance of success and can be cheaper.

Nova is an undersized rocket for launching crew. For example, Soyuz is one of the small crew capsules currently flying and it weighs 7 tons. This is beyond Nova’s 5-ton partially-reusable LEO capacity and makes it very difficult to create a crew capsule even if it is integrated into an existing spacecraft with maneuvering capabilities. A Cargo Capsule is more reasonable, but it may be unlikely that there will be sufficient demand for Stoke Space to pursue this. In his interview with NASASpaceflight, Andy Lapsa - Stoke Space CEO and cofounder - was asked about crew rating Nova and said this: “I actually expect that we’ll see a bimodal distribution in the optimal size of the vehicles. One of them is designed for satellites and one of them is probably bigger designed for humans with life [support] systems.”

Small Size if a Feature

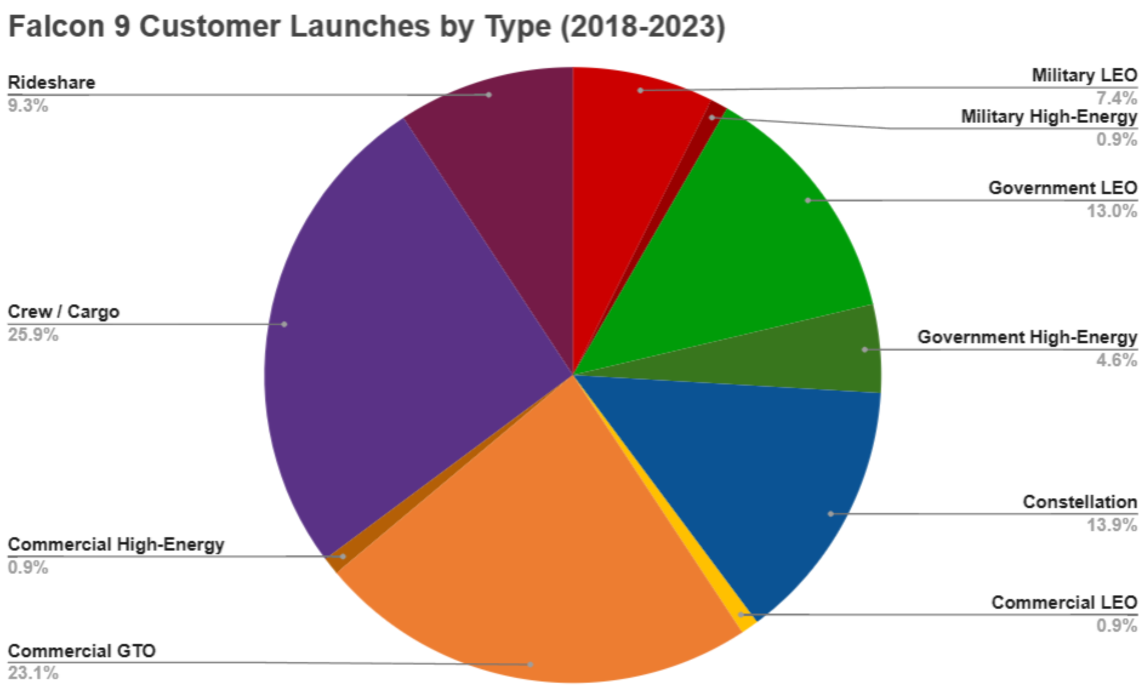
The fully reusable architecture allows for a high flight rate which can bring down the cost per launch and allows for rapid iteration.

Flight rate is one of the primary factors behind falling launch costs apart from vehicle architecture. Launching many multiples more than your competitors with a smaller fully-reusable vehicle allows for an accelerated process of lowering costs. Contrast this with Neutron, which will likely launch less than Falcon 9 with a second stage of similar complexity. This suggests it will be much harder for Rocket Lab than Stoke to compete with the Falcon 9.

Launching constellations would provide an excellent opportunity for Stoke as they could substantially increase launch cadence as Starlink did for Falcon 9. Constellation satellites are payloads that can accept more risk than other more expensive, one-off payloads. This means they can be launched on riskier life-leading boosters and upper stages. For several years, the life-leading Falcon 9 boosters only launched Starlink satellites. After years of testing and iteration SpaceX’s customers became comfortable with launching on these boosters. SpaceX’s current life-leading booster, B1060, recently launched the Odysseus IM-1 lunar landing mission on its 18th flight.

SpaceX iterated on Falcon 9’s design for years and refined it through the process of launching Starlink satellites on life-leading boosters. Thus far, Stoke has embraced iteration and hardware-rich development through the development of Nova’s upper stage and the first stage engine. Launching constellations will allow Stoke to used the same spirit of iteration, hardware-rich development, and pushing hardware to its limits to refine the vehicle.

Competing Against Partially-Reusable Launch Vehicles



Vehicles like Nova are the perfect competition against partially reusable medium launch vehicles. This can be illustrated by seeing where Nova and Neutron will compete. First, the average mass of customer payloads on Falcon 9 is 3.7t (excluding Dragon). This is addressable both by Nova and Neutron. Second, the next largest market for Falcon 9 is GEO communications satellites. Neutron GTO capacity is supposedly to be 3-5 tons and because Nova is planned to be refilled in orbit, it will have a comparable GTO capacity to Neutron. Finally, both rockets will compete for launching constellations and because Nova is designed for full reusability, it can achieve an overall lower cost than Neutron.

Almost every payload that can be launched on Neutron can be launched on Nova. The major exception is LEO satellites that are >5 tons. This is a relatively small market as it is mostly US

Military payloads that are launched by SpaceX and ULA. It is unlikely that Stoke or Neutron will be competitive in NSSL Lane 2 without more changes to NSSL policy, but they will be compete in NSSL Lane 1.

Heavy-lift rockets such as New Glenn and Terran R are much harder to compete against with a vehicle on the scale of Nova. Larger rockets can pursue unique missions and are better suited for heavy GEO satellites and massive constellations. For example, Nova could not compete for HLS as a vehicle on the scale of New Glenn or larger is required. Furthermore, for extremely large constellations such as Starlink or Kuiper, a larger rocket is better suited due to the sheer amount of satellites needed. Nova will need to launch many times a day to compete with these rockets on Starlink-sized constellations.

Competing Against Fully-Reusable Launch Vehicles

Stoke's Nova is clearly an excellent architecture to compete against partially-reusable launch vehicles, but Nova's position against larger fully and rapidly reusable launch vehicles is far less clear. In the next 10-20 years we will enter the paradigm of fully and rapidly reusable launch vehicles. It will not be just Starship and Nova, but every new rocket will be fully and rapidly reusable. When every major rocket is fully and rapidly reusable, Nova will not have all the benefits it will enjoy against partially reusable launch vehicles over the next ten years.

Large constellations will most likely be launched on large rockets. The sheer number of launches required to put the same mass into orbit compared to a heavy-lift reusable rocket makes it unlikely that a Nova-sized vehicle will be able to compete. For example, Nova would have to be 30x cheaper than Starship to be competitive. When rockets enter an airline-like model, cost per kilogram will become a more important factor and the larger scale of Starship will become more of a competitive advantage.

While large constellations will not be addressable by Nova, smaller constellations will be. All constellations require satellites to be launched into unique orbital planes to achieve proper coverage of the Earth. This can not be done on a single launch because of the immense fuel requirements to change orbital planes. On the scale of Starship, this means each launch may not carry enough satellites to be economical. For example, each Falcon 9 Iridium launch carried 10 satellites which weighed ~6.6t. This scale of constellation or smaller (Eg. Blacksky, Synspective, etc.) are well suited for a Nova-sized vehicle.

Aside from constellations, Nova may become the most economical small sat launcher. Its small size and full reusability will allow it to launch small satellites at a lower cost than any other vehicle, including Electron. Furthermore, it may not be economical to develop a smaller fully reusable small sat launcher due to the upfront development costs. A rocket designed for only small satellites will not be able to launch constellations or 1-5 ton satellites, which comprise the majority of the launch TAM. So, even if a smaller rocket is >50% cheaper per launch, it will take a very high number of flights to amortize the development cost. In 10-15 years Nova may begin launches more small satellites than any other vehicle.

In a world of fully and rapidly reusable rockets, Nova may be similar to Electron today. It will have a niche of payloads that are well suited for its size and cost. This includes small satellites, small constellations, and <5t satellites like those currently launching on Falcon 9.

Not Many Organizations Can Pursue Full Reusability

Current and upcoming rockets that are aiming to launch constellations are all partially reusable medium/heavy lift launch vehicles (excluding Starship, which is covered in the section above). A partially reusable rocket is the most reasonable thing for a slightly risk-averse company to do. For companies like Rocket Lab or ULA, it is very difficult to pursue a fully and rapidly reusable rocket because it is an unproven market and has never been done before which makes it a very hard sell to investors. This is especially true when you consider the development cost of such a program: ~$250M for Neutron vs. ~$5B for Starship.

It takes a level of audacity to push for such a program that will create its own market. SpaceX has done this again and again with Falcon 9, Starlink, and now with Starship & Mars. When existing stakeholders are not willing to take the risk, a more conservative approach must be taken that is less likely to succeed against a sufficiently capable competitor. We’ve seen this in the past with the SLS and other legacy launch companies like ULA.

A clear vision and path are required to pursue a fully and rapidly reusable rocket - along with immense amounts of capital. SpaceX achieved this under Elon Musk with his own fortune, NASA contracts, and venture capital. Stoke will follow a similar path, pursuing a goal that is decades away and beating the competition along the way because of this forward vision.

Conclusion & Predictions

A fully and rapidly reusable launch vehicle on the scale of Stoke’s Nova is perfectly sized to compete against existing and under-development partially reusable launch vehicles. It is well suited to launch constellations, LEO satellites, small & medium GEO satellites, dedicated small satellites, and small sat constellations. If full reusability is achieved with Nova, it will take significant market share over the next 5-10 years.

Over the next decade, we will start to enter the paradigm of fully and rapidly reusable rockets. These rockets will be able to launch all payloads at a lower cost than all existing vehicles and alleviate the current mass and volume limitations. The lower cost will allow for new markets to emerge and for a new paradigm of satellite design. K2 Space is already preparing for this by developing satellites that are very heavy and have a large volume. This increases launch costs but decreases the cost to build the satellite through using cheaper (but heavier) parts. There will come a point in the next 10-20 years when this becomes more economical than the current paradigm of satellite design, at which point a paradigm shift will occur.

Nova’s market this decade will be very different than the following decades. Before we firmly enter the paradigm of fully and rapidly reusable rockets and have payloads well suited to it, Nova will be able to compete against partially reusable rockets. After we enter this new paradigm, Nova will go the way of Electron and become more of a niche vehicle.

It is possible that Stoke will die brutally like Astra before it achieves low cost and high cadence with Nova. This would be immensely unfortunate because Nova is the coolest rocket ever developed, tied with Starship. However, even in such a depressing scenerio, a vehicle of similar scale to Nova will be developed in the future to adequately serve the <5t satellite market.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Comparing Demand for Firefly's Alpha vs. Electron

Feb 16, 2024 • Christopher Kalitin

Firefly's Alpha and Rocket Lab's Electron are both in a similar mass class of launch vehicles capable of catering to the small satellite launch market, but because of the differences in capability and price they are not in direct competition for most payloads.

Yesterday I was interviewed by Dave G about the small satellite launch market and we discussed Firefly's Alpha briefly. I felt the need to write this blog post to effectively convey my thoughts and show the information that made me come to my conclusions.

Firefly Alpha Planned Launches

Firefly's Alpha rocket is capable of delivering 1,030 kg to LEO while Rocket Lab's Electron does 300 kg to LEO. Due to Alpha's larger size it is more expensive than Electron and the primary feature of rockets in this class is launching payloads too massive for smaller rockets. This can be seen in Alpha's planned launches (from wikipedia).

1. NASA ELaNa Rideshare mission (11 satellites)
2. Commercial Rideshare mission (With orbital tug)
3. Satlantis Satellite (first launch, details unknown)
4. NRO Satellite (Classified US Military Payload)
5. ESO SAR 1 Satellite (~200kg+)

From this list of upcoming Alpha launches there appears to be several reasons for payloads to fly on Alpha instead of Electron with mass requirements being the most common. This presents few opportunities for competition.

First, the NASA VCLS (Venture Class Launch Services) rideshare mission contracts have been given to numerous launch providers as a way of supporting development of new launch vehicles. There isn't space for competition between Firefly and Rocket Lab here as the goal of the program is to give new launch vehicles extra funding. Electron is a mature vehicle and is not eligible for these launches anymore.

Second, the Commercial rideshare missions will use Firefly's Elytra orbital tug to deliver the payloads in an exact orbit. This is the first flight of Elytra and on it they will test all the systems. They are dispensing commercial small satellites and proving their capabilities for future NRO missions.

Thirdly, although there is little information available about it, judging from EOS Data Analytics' previous satellite and extrapolating forward, the EOS SAR 1 satellite may to be too heavy to launch on Electron. Furthermore, the CEO of EOS Data Analytics is Max Polyakov, a co-founder of Firefly.

Fourth, Satlantis is a young company so they may have not been able to get on Electron's manifest soon enough. All Electron launches in 2024 are booked and many are already scheduled in 2025. Furthermore, SpaceX Transporter missions have a 2 year wait. This shows merely having a functional launch vehicle - even one that costs 2x more than Electron - is enough to get awarded contracts when there are customers who can't afford to wait two years to launch their payload. Also, their satellite may be too massive to launch on Electron, we will know once it launches in 2024.

Finally, the NRO responsive space mission is part of a contract between the NRO and Firefly. The goal of this contract is to provide the capability to launch a payload for the NRO at a moment's notice. In 2023 Firefly demonstrated this capability when they launches the Vitcus Nox satellite within 27 hours of the launch order. Another requirement for this contract appears to be that it must take place in the US as it is a military payload that must launch on short notice. Rocket Lab only has one Electron launch pad in the US, so staying ready for a launch at all times may interfere with commercial payloads.

Competition between Alpha and Electron

With the context of Firefly’s planned launches we can consider the opportunities for competition between Firefly and Rocket Lab for these launches.

If the commercial rideshare mission is too massive to be launched on a single Electron, two launches could have been used. This would make the cost nearly equal between both launch providers and schedule becomes a larger concern. Because Electron is all booked for 2024 this gives Firefly and advantage until Rocket Lab increases cadence. Furthermore, with SpaceX transporter mission being booked for the next two years, Firefly was in a very good position to launch this mission.

If Electron was a larger vehicle it may have been able to launch the EOS SAR 1 satellite. However, the development costs associated with this make it not economical for Rocket Lab to pursue such a capability. Marginally increasing payload capacity is much less useful than increasing cadence or partial reusability.

With increased cadence and a smaller backlog for Electron, Rocket Lab may have gotten the opportunity to launch the Satlantis satellite. This is assuming it is not too heavy for Electron which is currently unknown. Rocket Lab has always been interested in increasing cadence and with current trends Electon may be demand limited within 2-3 years.

Schedule and payload requirements may have limited Rocket Lab’s ability to compete for the NRO responsive space contract. Rocket Lab’s first launch from the US was in early 2023 and the contract was awarded in August 2023. As mentioned above, Rocket Lab can get more use out of the pad when launching commercial payloads than keeping it primed for the NRO. So, for this contract to be a net gain to Rocket Lab they may have needed another launch pad in Virginia. Space is limited on Wallops Island and Rocket Lab had limited time which led to the company not being able to successfully compete for this contract. If another similar contract arises in the future it will be very interesting to follow and observe if Rocket Lab is in a better position to compete for it.

Overall, the only planned Alpha payload that reasonably could have flown on Electron is Satlantis’ satellite and possibly the rideshare mission. This shows the different niches both rockets are optimized to compete for.

Firefly does not have a vehicle well-suited to compete for payloads currently launching on Electron. The cost to launch Firefly’s Alpha is $15M while Electron costs ~$7M. With the 200% difference in cost, satellite operators are highly incentivized to launch on Electon. Furthermore, the average mass of payloads launched on Electron is ~116kg. The primary reason for a satellite that can launch on Electron to use Alpha is likely schedule considerations and Firefly may have gotten the Satlantis launch because of this. This schedule constraint will likely alleviate in the future as cadence increases.

In conclusion, the larger size and higher cost of Firefly's Alpha rocket makes it well-suited for different payloads than Electron. This niche of payloads will likely increase in number as the commercialization of space continues and this growth will offset potential competition from smaller and cheaper rockets.

Edit July 4 2024: This analysis didn't include small sat constellations, I've written more on this here.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Analysing the Dedicated Small Sat Launch Market

Feb 10, 2024 • Christopher Kalitin

In the last 6 years, Rocket Lab's Electron dedicated small satellite launches and SpaceX's Transporter rideshare missions have become regular occurrences. This has posed a problem many have struggled to answer by my standards: Why would any satellites be launched on a small rocket when rideshare missions are cheaper?

Jonathan McDowell Data					Orbit			Small Sat Type		
Launch Type	#Launc	LV_Type	Flight	Mission	Apogee	Perigee	Inclincation	Is Small Sat	Type	Sat Type
Commercial	2018-010	Electron	Still Testing	Dove/Lemurs	500 km	300 km	83°	Yes	Rideshareable	Earth Imaging
Government	2018-016	SS-520	Tasuki	TRICOM-1R	1,500 km	180 km	30°	Yes	Government	Earth Imaging
Commercial	2018-088	Electron	It's Business T	CICERO 10/Le	420 km	210 km	85°	Yes	Rideshareable	Earth Imaging
Government	2019-015	Vega	PRISMA	PRISMA	625 km	625 km	98°	Yes	Government	Earth Observa
Government	2019-016	Electron	Two Thumbs U	R3D2	425 km	425 km	40°	Yes	Military	Military
Military	2019-026	Electron	T.A.Funny Loo	STP 27RD	500 km	500 km	40°	Yes	Military	Military
Commercial	2019-037	Electron	Make It Rain	BlackSky Glob	500 km	490 km	94°	Yes	Unique Plane	Earth Imaging
Commercial	2019-054	Electron	Look Ma No Ha	BlackSky Glob	500 km	490 km	94°	Yes	Unique Plane	Earth Imaging
Commercial	2019-069	Electron	As The Crow F	Palisade	1,230 km	1,220 km	88°	Yes	Kickstageable	Earth Imaging
Commercial	2019-084	Electron	Running out of	ALE-2	400 km	410 km	97°	Yes	Rideshareable	Tech Demo
Military	2020-007	Electron	Birds of a Feat	NROL-151	610 km	590 km	71°	Yes	Military	Military
Military	2020-037	Electron	Don't Stop Me	NRO/ANDESITE/I	590 km	570 km	98°	Yes	Military	Military
Military	2020-046	Minotaur IV	NROL-129	NROL-129	580 km	570 km	54°	Yes	Military	Military
Commercial	2020-060	Electron	I Can't Believe	Sequoia	525 km	525 km	45°	Yes	Unique Plane	SAR
Commercial	2020-061	Vega	SSMS	SSMS	460 km	460 km	94°	Yes	Government	Rideshare
Commercial	2020-077	Electron	In Focus	CESAT-IIB/Flo	350 km	340 km	97°	Yes	Unique Plane	Earth Imaging
Commercial	2020-085	Electron	Return To Sen	Dragracer/BRC	480 km	470 km	98°	Yes	Rideshareable	Tech Demo
Commercial	2020-098	Electron	The Owl's Nigh	Strix-Alpha	510 km	490 km	97°	Yes	Rideshareable	SAR
Government	2021-002	LauncherOne	Elana XX	-	400 km	400 km	61°	Yes	Rideshareable	Rideshare
Commercial	2021-004	Electron	Another One L	BIU GMS-T	1,200 km	1,200 km	90°	Yes	Kickstageable	Earth Observa
Commercial	2021-023	Electron	They Go Up Sc	Global-5/Paths	460 km	450 km	42°	Yes	Unique Plane	Earth Imaging
Military	2021-051	Pegasus XL	Odyssey	TacRL-2	500 km	500 km	50°	Yes	Military	Military
Military	2021-052	Minotaur I	USA 316/317/3	NROL-111	600 km	580 km	96°	Yes	Military	Military
Government	2021-058	LauncherOne	STP-27VPA	Tubular Bells I	480 km	480 km	61°	Yes	Military	Military
Military	2021-068	Electron	It's a Little Chi	Monolith	600 km	600 km	37°	Yes	Military	Military
Government	2021-102	Epsilon	RAISE-2	-	560 km	560 km	98°	Yes	Military	Tech Demo
Commercial	2021-106	Electron	Love at First In	Global-14/Glob	460 km	450 km	42°	Yes	Unique Plane	Earth Imaging
Military	2021-108	Astra Rocket 3.3	Rocket 3.3	STP-27AD2	440 km	510 km	86°	Yes	Military	Military
Commercial	2021-120	Electron	Data With Dest	Global-16/Glob	460 km	450 km	42°	Yes	Unique Plane	Earth Imaging
Government	2022-003	LauncherOne	STP-27VPB	Above The Clouds	510 km	500 km	45°	Yes	Rideshareable	Rideshare
Commercial	2022-020	Electron	The Owl'sNigh	Strix Beta	550 km	540 km	96°	Yes	Rideshareable	SAR
Commercial	2022-026	Astra Rocket 3.3	Rocket 3.3	Spaceflight Ast	525 km	525 km	98°	Yes	Rideshareable	Rideshare
Commercial	2022-034	Electron	Without Missic	Global-18/Glob	450 km	460 km	53°	Yes	Unique Plane	Earth Imaging
Commercial	2022-047	Electron	There and Bac	E-Space Demc	520 km	520 km	98°	Yes	Rideshareable	Tech Demo
Government	2022-070	Electron	CAPSTONE	CAPSTONE	300,000 km	200 km	28°	Yes	Unique Plane	Other
Government	2022-074	LauncherOne	STP-S28A	Straight Up	400 km	380 km	45°	Yes	Military	Military
Military	2022-079	Electron	Wise One Lool	NROL-162	620 km	620 km	40°	Yes	Military	Military
Military	2022-091	Electron	Antipodean Ac	NROL-199	620 km	620 km	70°	Yes	Military	Military
Commercial	2022-113	Electron	The Owl Sprea	Strix-1	560 km	550 km	98°	Yes	Unique Plane	SAR
Commercial	2022-122	Firefly Alpha	To The Black	-	300 km	300 km	137°	Yes	Rideshareable	Rideshare
Commercial	2022-127	Electron	It Argos From	GAzelle	767 km	750 km	98°	Yes	Kickstageable	Earth Observa
Government	2022-147	Electron	Catch Me If Yo	MATS	585 km	585 km	98°	Yes	Kickstageable	Earth Observa
Commercial	2023-011	Electron	VA is for Laun	Hawk 6	550 km	540 km	41°	Yes	Unique Plane	SAR
Government	2023-019	SSLV	EOS-07	EOS-07	440 km	430 km	93°	Yes	Government	Tech Demo
Commercial	2023-035	Electron	Stronger Toget	Capella 9/10	575 km	565 km	44°	Yes	Unique Plane	SAR
Commercial	2023-041	Electron	The Beat Goes	Global-19/Glob	464 km	451 km	42°	Yes	Unique Plane	Earth Imaging
Government	2023-044	Shavit 2	'Ofeq-13	'Ofeq-13	500 km	500 km	140°	Yes	Military	Military
Commercial	2023-062	Electron	Rocket Like A	Tropics SV05/S	550 km	550 km	32°	Yes	Unique Plane	Earth Observa
Government	2023-072	Nuri	NEXTSAT-2	-	540 km	550 km	98°	Yes	Government	Tech Demo
Commercial	2023-073	Electron	Coming To A S	Tropics SV03/S	550 km	550 km	32°	Yes	Unique Plane	Earth Observa
Commercial	2023-100	Electron	Baby Come Ba	Electron 39 Ric	1,000 km	1,000 km	99°	Yes	Kickstageable	Earth Observa
Commercial	2023-126	Electron	We Love The N	Capella 11 Aca	648 km	640 km	53°	Yes	Unique Plane	SAR
Government	2023-188	ADD TV2	S-STEP	-	650 km	650 km	50°	Yes	Government	SAR
Commercial	2023-196	Electron	The Moon God	QPS-SAR-5	587 km	577 km	42°	Yes	Unique Plane	SAR
Commercial	2023-202	Firefly Alpha	Fly the Lightni	Tantrum	Military			Yes	Military	Military

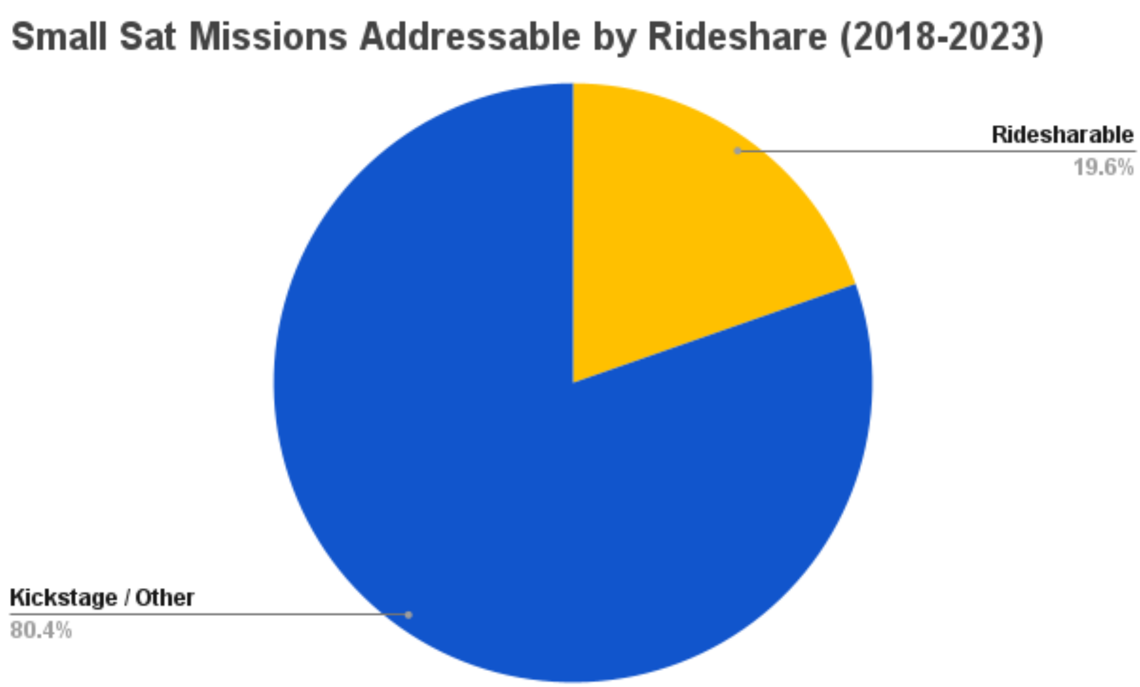
[Expanded Source Data Image](#)

To access the source data from the spreadsheet: copy the spreadsheet and in Cell BC2 filter to

only small sat launches.
https://docs.google.com/spreadsheets/d/1VOgRbnAsQZdGIPoemRj5ApSLk_jxGanNliWEPnBB3p4/edit?usp=sharing

More, better, and more accurate charts here: https://drive.google.com/drive/folders/1IAbB-Ydgv3udhIhvfdKp_dWnxUshMqmc?usp=drive_link

Rideshareable Satellites



In the past 6 years (2018-2023 inclusive), 55 dedicated small sat launches have occurred excluding Chinese rockets. By my criteria, 11 of these missions could have been completed by launching on a commercial rideshare mission. However, five of these appear to prefer a dedicated launch and the remaining six are more well-suited for a rideshare mission on a medium launch vehicle.

These missions are (Rocket, Payload):

1. Electron - Dove/Lemurs
2. Electron - CICERO 10/Lemurs
3. Electron - ALE-2
4. Electron - Dragracer/BRO/SpaceBEE
5. Electron - Strix-Alpha
6. LauncherOne - Rideshare
7. LauncherOne - Rideshare
8. Electron - Strix-Beta

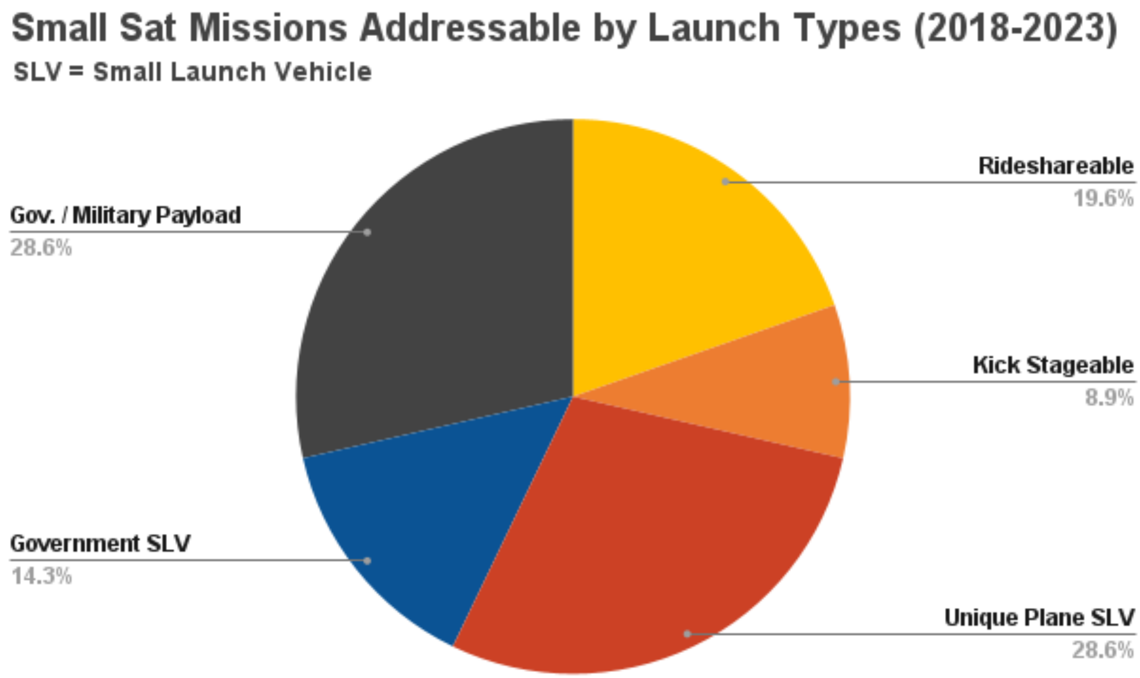
9. Rocket 3 - Rideshare
10. Electron - E-Space Demo
11. Alpha - Rideshare

The rideshare, Planet Labs Dove, and CICERO 10 launches are well suited for a rideshare mission on a medium launch vehicle (MLV). The rideshare missions are all risky flights on unproven launch vehicles (Astra Rocket 3, Firefly Alpha, Virgin Orbit LauncherOne). The reason these payloads flew on a small launch vehicle (SLV) is likely due to the lower cost of flying on a risky rocket. For example, many low-cost Government / Military rideshare payloads flew on Virgin Orbit's LauncherOne. The Dove and CICERO 10 launches are also well suited to rideshare missions because an exact orbit is not required. Planet Labs have been launching payloads on rideshare missions since the first flight of Antares.

The remaining five missions are all tech demonstration satellites and all require a commonly-used Sun Synchronous Orbit (SSO). This is a ~500km orbit at an inclination of 98 degrees that allows a satellite to pass over locations on Earth at regular intervals. This is the orbit every SpaceX Transporter rideshare mission has gone to, but there are more considerations than the end destination.

The reason for launching on a dedicated small rocket instead of a rideshare mission is likely schedule requirements. Rideshare missions only launch a couple of times per year and are scheduled by the launch provider. Small sat operators have much more control over launch date and orbital parameters when flying on a dedicated launch vehicle. This is often cited as the primary reason for companies choosing a dedicated launch over a rideshare mission. However, these five commercial schedule-constrained missions account for only 9% of the small sat launches in the last 6 years.

Kick Stageable Satellites



Only five of the previous 55 small sat launches are suitable for launching on a rideshare mission and using a kick stage. A Kick Stage is a small stage on top of the rocket that can change the orbit of the payload so it is delivered to the specific orbit that is required.

Missions (Rocket, Payload):

1. Electron - Palisade
2. Electron - GMS-T
3. Electron - GAzella
4. Electron - MATS
5. Electron - Rideshare

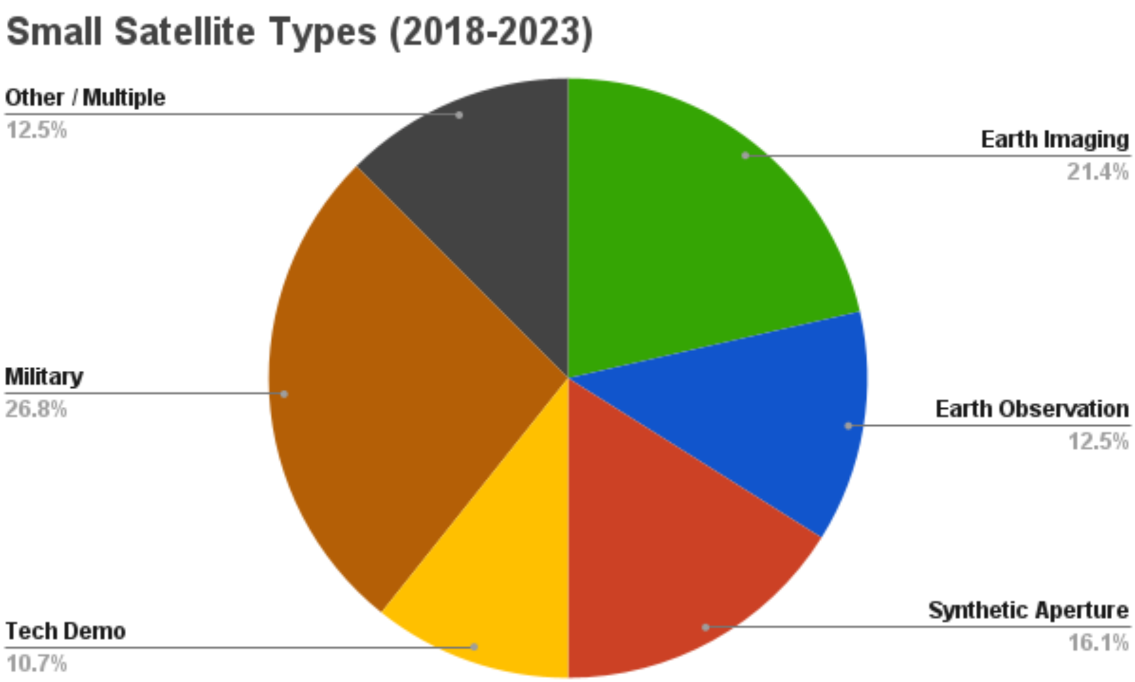
Each of these five launches have satellites that require unique orbits around the Earth and cannot use the default SSO orbit that many rideshare missions arrive at. The highest orbit is that of Paliside at 1,200km and an 88-degree inclination as of February 9 2024. The lowest orbit is MATS at 590km and a 98-degree inclination, this launch was also Rocket Lab's second attempt at catching a booster.

All the launches mentioned above have inclinations that are within several degrees of the 98 degrees that is common for rideshare missions. Major inclination changes require immense amounts of fuel to complete, so more than a couple of degrees of inclination change is not reasonable to be completed with a kick stage.

Kick stages are very useful for achieving a target orbital altitude. In the past, this has been a capability only achieved through dedicated launch or complex satellites that have propulsion built in.

Only ~10% of small sat launches can be addressed by a kick stage. For all the news around them, this appears to be a fairly low number, but this is only small kick stages like Impulse's Mira. The larger market (especially economically) for kick stages may be as a GEO delivery mechanism.

Dedicated Launch Required Satellites



Sixteen small sat launches of the last 55 are in a category that is not addressable by rideshare or kick stages. These are satellites that require a unique orbital plane to take their place in a constellation. To properly cover the world with a satellite constellation the satellites must be placed into precise orbital planes so that there are no locations on the surface that lack coverage. For example, Starlink Shell 1 consists of 72 orbital planes with 22 satellites in each. Satellites cannot be moved between orbital planes due to a similar reason as inclination changes, it requires an immense amount of fuel.

The requirement for a unique orbital plane and the immense difficulty of addressing this market with a kick stage makes it the perfect market for small launch vehicles.

All of these launches carried satellites that conduct Earth observation:

1. 7x Blacksky (Imaging)
2. 3x Capella (Synthetic Aperture Radar)
3. 2x Tropics (Weather Observation)
4. 4x others (mostly Synthetic Aperture Radar)

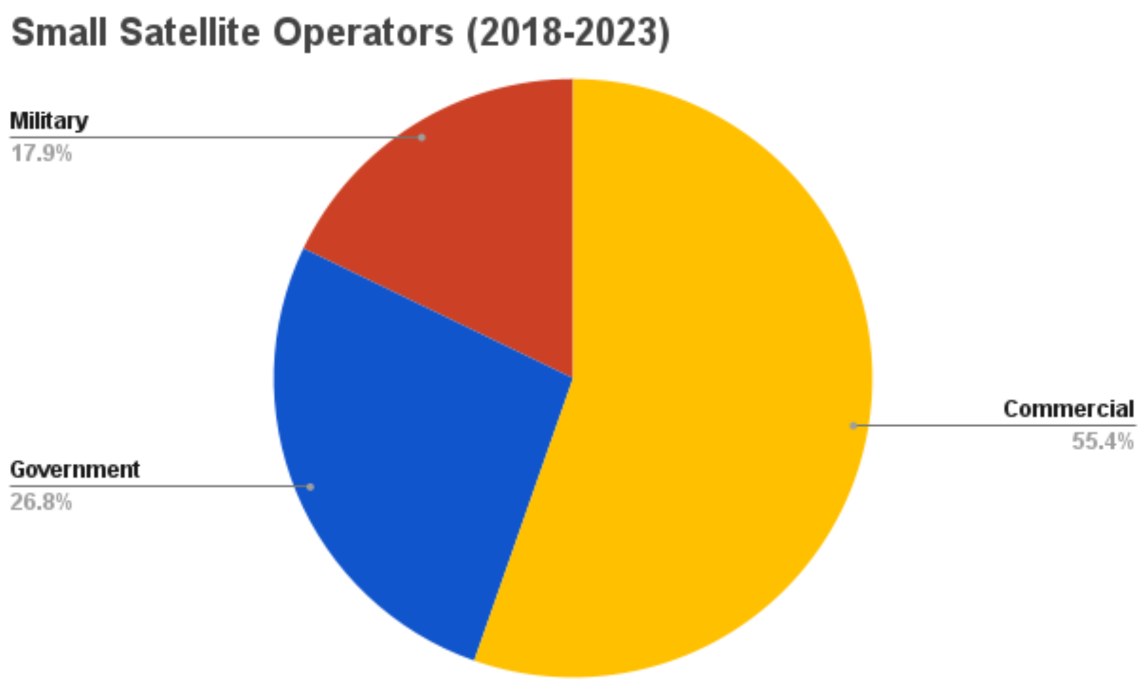
These Earth observation constellation satellites all require launching to a specific orbit on a small launch vehicle, but not all constellations do. The prime example is Planet Labs. Their constellation is used for Earth Imaging but they have flown many times more on rideshare missions than small sat launches. Planet Labs solved the dedicated launch problem through the sheer scale of their constellation. They currently operate 150 satellites and have launched 462 in total. Because of the size of this constellation, the number of launches required, and eased requirements on orbital planes, Planet Labs is able to forego the need for dedicated launches.

Rocket Lab is currently targetting 22 launches in 2024 and the majority of these are for Earth Observation satellites:

1. 3x Spire
2. 2x NASA Prefire
3. 5x Kinéis
4. 2x Capella
5. 3x Strix
6. 2x Hawk
7. 5x Blacksky
8. 2x Non-Earth Observation

All 16 of the launches that required a unique plane in the last 6 years were launched by Electron. Furthermore, 22 of the 24 currently planned Electron launches are for constellation satellites that require unique orbital planes. These are launches that can only be done with a small satellite launch vehicle and will be the primary business of small launch providers for the coming years.

Government and Military Satellites



The Government and Military are not rational consumers. In the commercial world, everything makes sense because economics drives it all, this is not true of the government.

24 of the previous 55 small sat launches were paid for by National Space Agencies or Militaries. 11 of these launches used a government-developed launch vehicle (Rocket, Customer):

1. SS-520 - Japan
2. Vega - Europe
3. Minotaur IV - USSF
4. Vega - Europe
5. Pegasus - USSF
6. Minotaur I - USSF
7. Epsilon - Japan
8. SSLV - India
9. Shavit 2 - Isreal
10. Nuri - South Korea
11. GYUB-TV2 - South Korea

Many countries are interested in developing their own launch capability and do so with a government-led program. This often leads to achieving the goal of developing a rocket, but not often an economical one. Because the goals of these programs are more aimed toward increasing the technological sophistication of a country and not solely to create a cheaper and reliable rocket, these programs often struggle against privately developed rockets and require government subsidy to stay afloat.

This is all to say the above 11 launches do not compete in the same arena as the rest of the commercial small satellite launches. For example, the Vega rideshare mission is part of an ESA program to launch small satellites. The ESA perfers launching payloads on their own rockets due to the geographical return policy, and appear to only launch on non-european rockets begrudingly. With the commercialization of spaceflight, this paradigm - that has been present since the V2 - will go away in favour of cheaper and more reliable commercial rockets.

The remaining 13 government/military launches were all done for US customers by purchasing a launch on a commercial provider's rocket. This method opens competition among launch providers to provide the best service.

Launches (Rocket, Customer):

1. Electron - NASA
2. Electron - USSF
3. Electron - USSF
4. Electron - USSF
5. Electron - USSF
6. LauncherOne - USSF
7. Electron - USSF
8. Rocket 3 - USSF
9. Electron - NASA Capstone
10. LauncherOne - USSF
11. Electron - USSF
12. Electron - USSF
13. Alpha - Lockheed Martin

The US Military almost exclusively launches its satellites on dedicated launches. This is to maintain national security by having tight control over who has access to the satellite. The requirement for dedicated launch means these satellites cannot be launched on rideshare missions and will continue to be a large part of the small satellite launch market even when ridesharing & kick stages offer significant cost savings.

The NASA Capstone mission is not beholden to the same anti-espionage requirements. The reason it had a dedicated launch was Electron's ability to provide the Photon satellite bus / kick stage to reach a trajectory that passes the Moon. Even though a kick stage was used, this mission is not addressable by a rideshare mission because launching to the Moon requires the satellite to be in the same orbital plane as the moon and no rideshare missions have ever gone to this orbit.

Cost Estimates

Determining the price of a dedicated small satellite launch is straightforward as we can use the cost of an Electron launch, ~$7M.

The cost of a rideshare mission can be derived from what SpaceX charges for Transporter missions. They currently charge ~$5,500 / kg for a satellite on a transporter mission. The average mass of the payloads launched on Electron is 116kg. Multiplying these two numbers gives $638,000 as the cost of launching on a rideshare mission. This is 11x cheaper than a dedicated launch.

Determining the price of using a kick stage on a rideshare mission is more difficult because there are no publicly available prices for this service. We can estimate the price of launch by using Impulse Space's Mira kick stage. Mira weighs 250kg and can carry up to 300kg of payload while having 500m/s of delta-v. Using the average electron payload mass, the total mass of the kick stage and satellite is 366kg. This costs $2M to launch on a SpaceX Transporter mission, I will assume $2.5M because this is one of the larger payloads on a Transporter mission and there may be extra considerations around payload volume.

The cost of manufacturing the kick stage itself is unknown, but we can use Rocket Lab's Photon as an analogue. The Electron and Photon for NASA's Capstone mission cost $10M in total. We already know the cost of an Electron launch is ~$7m, so Rocket Lab appears to have charged an extra $3M for Photon.

This gives a total of $5.5M for a rideshare + kickstage launch vs. $7M for a dedicated launch.

Furthermore, there is more room for cost declines in using rideshare + kick stages compared to dedicated launches. SpaceX's margins on Falcon 9 are >50% as launch costs are ~$20M and they make ~$45M from Transporter missions. Rocket Lab's margins are ~22% on Electron. There is room for improvement in Rocket Lab's margins as they reuse boosters and increase scale, however, they are currently planning to reuse only ~50% of boosters and only increase scale ~2x.

This shows that the gap in cost between rideshare + kick stages mission and dedicated launches will stay at around $1M - $2M for the foreseeable future. The incentive for a payload to use a dedicated launch when a rideshare mission is possible appears to only be schedule considerations.

Summary

Around 20% of the existing dedicated small satellite launch market will go away with the increased cadence of rideshare missions and kick stages becoming a viable option. The market for dedicated launch has already been distilled through SpaceX transporter missions, with an increase in cadence and kick stages further distillation will occur. If rideshare missions and kick stages were ubiquitous in the past, I estimate ~50% of the small satellites launched in the previous 6 years that had the choice between rideshare/kick stage vs. dedicated launch would have flown on rideshare missions.

The 30% of satellites that require unique orbital planes will remain part of the dedicated launch market forever. It is too difficult to significantly change the orbital plane of a satellite even with a large kick stage, so, this market is only addressable by dedicated launch. This can already be shown through Rocket Labs' launch manifest. >90% of their planned launches for 2024 and 2025 fall into this category.

Military launches make up another 30% of the dedicated small satellite launch market. National security requirements preclude these launches from occuring on rideshare missions so this segment will not be threatened.

Government-funded launch vehicles make up the remaining 20%. These programs exist because of political incentives so it is difficult to determine if they will continue into the future or not. However, with the push for commercialization in space flight, it is possible this market will slowly evaporate as privately funded alternatives arise in their respective countries. This segment is not addressable by commercial providers, so it is not of major interest.

The dedicated small satellite launch market has already faced tremendous competition from rideshare missions. Since its inception, SpaceX's Transporter program has launched about 785 satellites. Electron has launched about 180 satellites. Rideshare missions already have 80% of the total small satellite launch market. Dedicated launch is a somewhat niche service for unique satellite constellations and government customers.

Around 30% of the market for dedicated small satellite launches addressable by commercial providers will disappear with better alternatives arising. However, this does not translate into a fall in the quantitative demand. Rocket Lab's launch cadence is planned to increase from 9 to 20 this year. This 200% growth in a single year will offset the decline from rideshare missions. The rise of small satellite constellations and economically productive uses of space will drive substantial growth in the dedicated small satellite launch market even taking into account the decline from rideshare missions kick stages.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. <a href="https://caseyhandmer.wordpress.com/2023/08/25/you-should-be-

working-on-hardware/">You should be working on hardware

Christopher Kalitin Blog

Analysing Neutron in the Commerical Satellite Launch Market

Jan 7, 2024 • Christopher Kalitin

Neutron is Rocket Lab's next-generation medium-lift rocket. In many ways, it looks and functions like Falcon 9 version 2 as it is designed from the ground up for reusability. This is most obvious with the booster. The fairings are built into the structure of the booster, the landing legs are built into strakes, and the Archimedes engine is designed to operate below maximum performance to optimize for reusability.

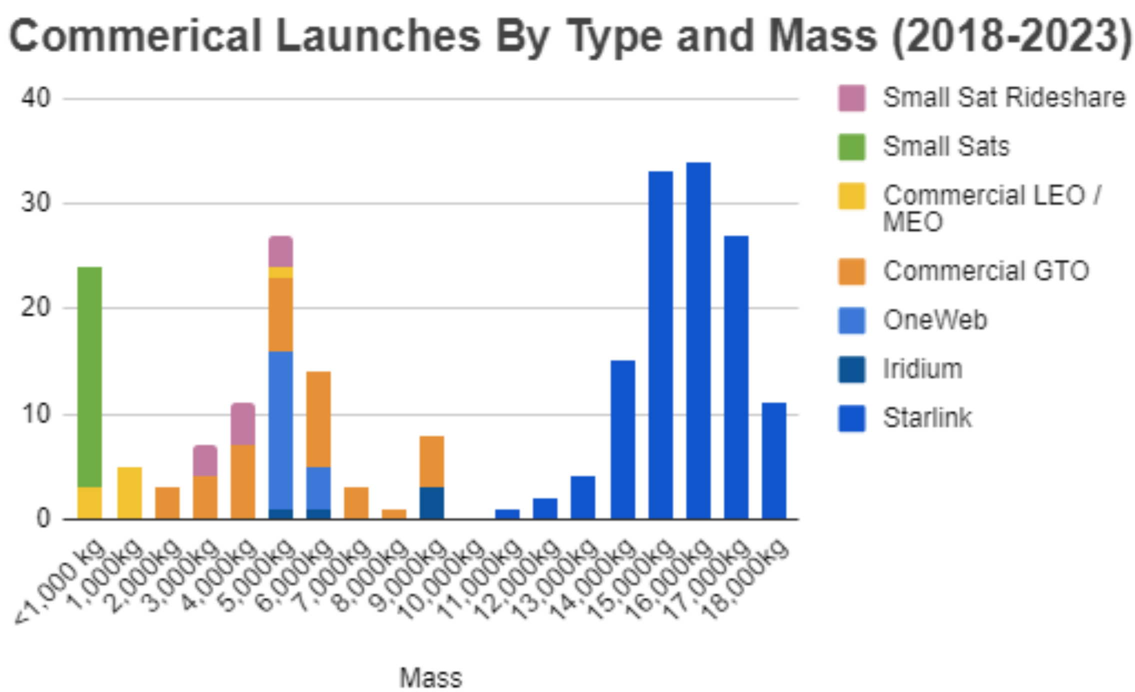
During the early development of Falcon 9, SpaceX was optimizing for the lowest probability of bankruptcy. With Neutron, Rocket Lab can optimize for the most competitive and efficient rocket.

However, Neutron is not a perfectly optimized rocket compared to its competition. In many ways the Falcon 9 and other future medium/heavy lift rockets out perform Neutron in its primary market of launching constellations. The volume of the fairing and payload mass are both lower than comparable rockets such as the Falcon 9, Terran R, and New Glenn and it is not optimized for high-energy orbits meaning it can't launch most GTO satellites.

Model this analysis is based on: https://docs.google.com/spreadsheets/d/1VOgRbnAsQZdGIPoemRj5ApSLk_jxGanNliWEPnBB3p4/edit?usp=sharing

More, better, and more accurate charts here: https://drive.google.com/drive/folders/1IAbB-Ydgv3udhIhvfdKp_dWnxUshMqmc?usp=drive_link

Legacy Commerical Communication Satellite Market



Updated Chart (Important! Click Me!)

The legacy commercial communication satellite market is dominated by GEO satellites. Between 2018 and 2023, 48 commercial satellites were successfully launched. 39 of these satellites were launched to GTO/GEO while only 9 were launched to LEO/MEO/SSO.

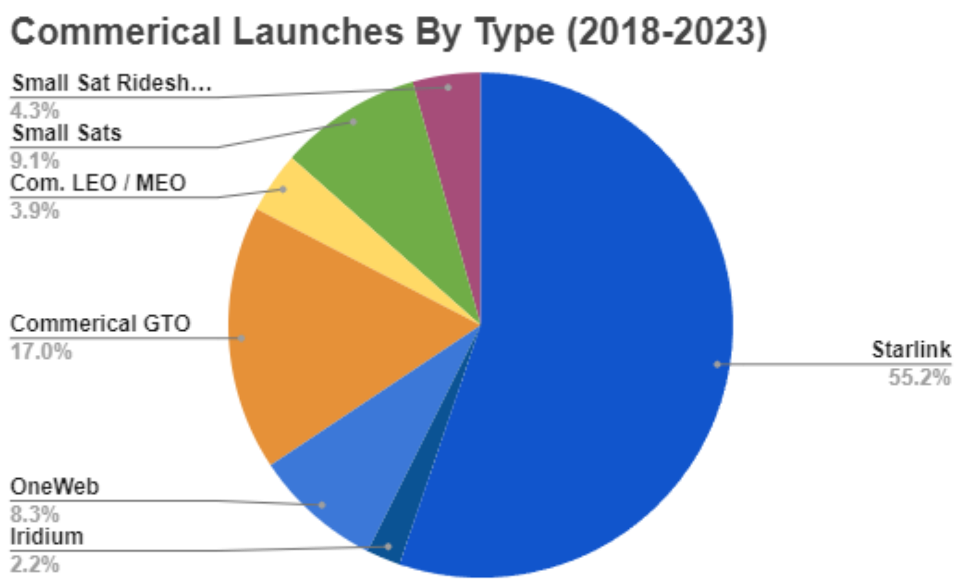
Neutron can only launch ~50% of these satellites and most require expending the booster. This was the primary commercial market for the Falcon 9 after the original CRS test launches. Before 2018, Falcon 9 conducted 6 LEO commerical satellite launches and 17 to GEO/GTO.

In the early days of Falcon 9 commerical launches it took market share from the Ariane 5. The Ariane 5 was optimized for high-energy orbits which was the majority of the market at the time.

Ariane 5 peaked at 7 launches per year. Compared to the 96 Falcon 9 launches in 2023 this is a relatively small market and shows that the new paradigm of constellations is far more important than the legacy market of commercial communication satellites. Clearly new rockets should be optimized for launching LEO constellations and not GEO satellites. This point is more of an indictment of Vulcan than Neutron. Even the majority of Vulcan's booked launches are for a LEO constellation.

Neutron will not take market share here and I don't believe Rocket Lab should put much effort into this as it is now a minor market compared to Constellations.

Constellations - The Next Paradigm



Constellations are the next paradigm in the commercial communication satellite market and what Rocket Lab claims Neutron is optimized for.

Constellations do not require high-energy orbits. Rather, they require high mass-to-orbit over multiple launches. So, there are no stringent minimum mass requirements for constellations as there are for GEO satellites. Total mass to orbit is the primary metric and this can be achieved by any rocket through increasing the total number of launches. Cost per launch per satellite is the metric that should be optimized for.

There are multiple LEO constellations waiting to be launched. Kuiper is the most well-known and there are many smaller constellation projects such as Iridium, OneWeb, Orbcomm, Intelsat, etc. Launch TAM in total until 2030 may be upwards of 10 billion dollars excluding Starlink.

Small Sat constellations are a minority of the market. Small sats can be launched on rideshare missions and do not require massive amounts of launches. To illustrate this point, Planet Labs currently has around 200 operational satellites in orbit and each Dove satellite weighs around 5kg. 200*5=1000kg. The entire constellation is only a couple of tons, if specific orbits weren't required, this could be done on a single Falcon 9.

Neutron's Competitive Position in Constellations

	A	B	C	D	E	F	G	H	I	J	K	L	M
1		**Falcon 9**	**Falcon Heavy**	**Neutron**	**New Glenn**	**Terran R**	**Vulcan VC0**	**Vulcan VC6**	**Ariane 5**	**Ariane 6 A64**	**MLV**	**Notes**	**Starship**
2	**Mass to Orbit**												
3	LEO Expendable	22.8t	63.8t	15.0t	55.8t	33.5t	N/A	N/A	20.0t	21.7t	16.0t		
4	LEO Reusable	18.4t	28.4t	13.0t	45.0t	23.5t	10.8t	27.2t	N/A	N/A	N/A		150.0t
5	GTO Expendable	8.3t	26.7t	5.0t	19.6t	8.3t	N/A	N/A	10.9t	11.5t	5.0t		
6	GTO Reusable	5.5t	8.0t	3.0t	13.0t	5.5t	8.4t	15.3t	N/A	N/A	N/A		30.0t
7													
8	**Fairing**												
9	Fairing Diameter	5.2m	5.2m	5.0m	6.4m	5.0m	5.4m	5.4m	5.4m	5.4m	5.0m		9.0m
10	Fairing Height	13.1m	13.1m	10.0m	17.8m	21.5m	21.3m	21.3m	17.0m	20.0m	9.0m		20.0m
11	Payload Volume	222.6m³	222.6m³	157.1m³	451.0m³	337.7m³	390.3m³	390.3m³	311.5m³	366.4m³	141.4m³		1017.9m³
12													
25	**Satellite Capability**	**<- Rough estimates from Mass & Volume**											
26	Starlink V2 Mini	21	26	15	36	27	12	31	23	25	18		160
27	Kuiper (est.)	28	30	20	60	36	17	42	31	33	19		57
28	Iridium	10	10	7	20	15	16	18	14	17	6		46
29	OneWeb	40	40	28	81	60	70	70	56	65	25		182
30													
31	**Rocket Info**												
32	Launch Cost	$60.0M	$150.0M	$40.0M	$70.0M	$55.0M	$110.0M	$110.0M	$178.0M	$115.0M	$70.0M		$100.0M
33	Launch / Year Limit	365	12	365	365	365	24	24	7	12	24		1825
34	$ / kg	$3,261	$5,282	$3,077	$1,556	$2,340	$10,185	$4,044	$8,900	$5,312	$4,375		$667
35													
36	**Cost / Satellite**												
37	Starlink V2 Mini	$2.9M	$5.7M	$2.7M	$2.0M	$2.0M	$8.9M	$3.5M	$7.8M	$4.6M	$3.9M		$0.6M
38	Kuiper (est.)	$2.1M	$5.1M	$2.0M	$1.2M	$1.5M	$6.6M	$2.6M	$5.8M	$3.5M	$3.7M		$1.8M
39	Iridium	$5.9M	$14.8M	$5.6M	$3.4M	$3.6M	$6.7M	$6.2M	$12.6M	$6.9M	$10.9M		$2.2M
40	OneWeb	$1.5M	$3.8M	$1.4M	$0.9M	$0.9M	$1.6M	$1.6M	$3.2M	$1.8M	$2.8M		$0.6M
41													
42	**Market Fit (2018-2023)**												
43	Commerical LEO / MEO	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	2 / Year	
44	**Commerical GTO**	***87.18%***	**100.00%**	***53.85%***	**100.00%**	***87.18%***	***87.18%***	**100.00%**	**100.00%**	**100.00%**	***35.90%***	8 / Year	
45	Small Sat Rideshare	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	4 / Year	
46	Constellation	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	~80 / Year	
47	**Total Market Fit**	**97.83%**	**100.00%**	**92.17%**	**100.00%**	**97.83%**	**97.83%**	**100.00%**	**100.00%**	**100.00%**	**89.13%**		
48													

Click here for expanded version of chart or source

Cost per launch per satellite is fundamentally what is being optimized when launching constellations, with responsiveness tied for first or in second place. Payload mass and volume are the primary determining factors for the number of satellites that can be launched. This is clear with Starlink as SpaceX was constrained by volume with v1 and now is constrained by mass on v2 mini.

Above you can see the cost per launch per satellite for current, under-development, and retired medium/heavy lift rockets for a few major constellations. Neutron must be priced far lower than other medium-lift rockets to be competitive at my price-per-launch estimates. If you'd like to use your own estimates, copy the spreadsheet and make your own adjustments.

Neutron will have to charge ~$40M per launch to be competitive on constellations. This is not a consideration for other larger LEO/GTO satellites as they are not mass-limited. You can see this when Falcon 9 launches ~5t LEO satellites which is far below the 18.4t max payload.

SpaceX's original Iridium deal was $492M for 8 launches which is $61.5M per launch. If Neutron carried out this contract, it would have to launch ~13 times for $38.7M per launch. Worse yet, Neutron is volume limited here as the 10 Iridium satellites are only 6.6t. With reusability, $40M per launch is possible, but at significantly lower margins than the Falcon 9 and potentially other medium launch vehicles in the future.

Neutron's improved partially-reusable design over Falcon 9 shows some promise for lower costs, but design is not the hard part. SpaceX produces 100 F9 second stages per year and Falcon 9's marginal cost per launch is ~$15M. It is unlikely that Neutron's second stage will be cheaper and with the increased competition I am doubtful that Neutron will exceed 50 customer launches per year before the paradigm of fully-reusable rockets arrives. These points apply to their competition as well. Economies of scale will be difficult to achieve with Neutron and as a hopeful future RKLB investor I don't see lower costs than F9 as likely.

Conclusion

In my recent video on updates to my RKLB valuation model, the primary piece of intelligent criticism I received was about my comments about Neutron. I've spent the last 2 days and 12 hours in total researching and creating a model of the commercial satellite launch market to improve my understanding.

"Constellations are what you want to be aiming for. I'm not sure Neutron is optimized to launch constellations."
I completely misunderstood the cost per launch component. My fundamental criticism of Neutron is that because of its lower size compared to other medium/heavy lift partially-reusable rockets, it will have to charge much less per launch to be competitive. However, with the smaller size of Neutron and Rocket Lab's proven ability to development launch vehicles for low cost, Neutron may be well positioned to achieve lower launch costs than competitors.

"[For Neutron to efficiently launch constellations] it will have to land at sea with low payload volume with low payload mass. This is not the best architecture I'd say, something like relativity's Terran R I'd be more bullish on."
I completely stand by this statement. New Glenn appears to be the best partially-reusable rocket to launch mega-constellations from a design perspective. RTLS + High payload mass + Large fairing volume makes an ideal 2050 rocket. Neutron is suboptimal on all three of these metrics.

Execution is fundamentally what you should be concerned with in rocket development programs. This can be seen in the chart above as the Falcon 9 is suboptimal and yet launches the vast majority of satellites - world class execution is far more important than anything else. In the video I missed this point entirely as I was focused on being pessimistic about Rocket Lab to combat my inherent optimism as can be seen in my valuation estimates.

Rocket Lab has a proven track record and I believe they will be able to execute on the Neutron program while bringing in billions in profits over its lifetime. However, Neutron remains a suboptimal rocket.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Patriotism is a Requirement

Jan 1, 2024 • Christopher Kalitin

I intend to become a citizen of the Land of Opportunity and do everything in my power to improve the nation. Albeit, until I find a place that aligns more closely with my ideals.

This is patriotism, being a net positive to your nation. I can't imagine living a net-zero or net-negative life, what's the point? Work to improve your life and the lives of your neighbors and family members. This is a fundamentally useful and positive life and everyone should strive for it.

Without a positive framework for living you fall into the trap of working for money or working for leasure. We should teach the kids to improve the future - pessimism is Death.

It appears religion used to serve this role. Christianity provides a good framwork for living: Love your neighbours, have faith, forive, don't be hypocritical, repent and ask forgiveness for your sins, etc. This sounds like a life I would like to live. Go full Christian!

Under athiesm you need to create your own framework. Here you can fall into traps of nihilism, hedonism, or other -isms. It's okay to just be happy! You don't need to have kids anyway, it's so hard! The future is bad anyway, why bother? Nothing we can do! Not sure of who you are, maybe you're actually a woman!

It is very difficult to create your own framework for living. Religion provides this from birth. In the absense of religion, you better hope your parents raise you well and set good examples!

Back to the point: Getting good people together who want to improve their society is an amazing thing.

This is the goal of good immigration policy. Let in the net-positives, keep out the net-negatives.

You need an efficient way to sort potential immigrants. This requires quickly judging applications and only allowing those who are accepted into your country. Quotas or lotteries miss the goal: only let in the good ones. A limit to the number of immigrants can lead to missing good people and an insecure border leads to bad people getting in.

Patriotism is a requirement for immigrants. You must want to improve the country you are moving to. Give every immigrant and 18 year old a US history test in order to be allowed to vote.

This is the system I want to live under. I want to live in a country I want to improve and move when I find a better one.

Without patriotism, you lose your culture and the ideals of your country. 40% of people in Vancouver are immigrants and with a fertility rate of 1.1 the future of the culture doesn't look good. There are uses for multiculturalism, but it is unfortunate to lose your culture. I was driving through Surrey 3 days ago, it was very Indian.

Again, we should aspire to get like-minded people together to build a great future for their society. Patriotism > Multiculturalism.

Like many political instituions, current immigration policy in the Land of Opportunity misses the goal.

Why do I have to lie in University Admissions essays? If I were honest I'd say: "I'm not writing your idiotic essay prompt, here are 30k lines of code I've written and straight As. This proves I will be a good Comp Sci student."

Maybe the solution is political. Javier Milel gives me some hope, but I doubt that could happen in the US. Maybe there is a deep state.

There was a time in history when a new nation caused the transformation of the world to democracy. Maybe we need Mars to do the same.

What happens if the US loses its fundamental beliefs (based on Christianity). Zuby and E believe if the West loses Christianity it is screwed. Maybe they have a point. A group of people without common fundamental beliefs is not very useful to the future. Acclerate!

This has already been a ramble and I've been watching George Hotz streams from hours while being sick and groggy.
So, I'd like to use the end of this post to give a message my future self:

DON'T BE A SELLOUT!

Subscribe

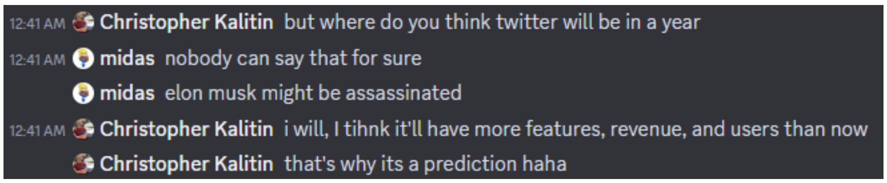
CKalitin
x.com/CKalitin

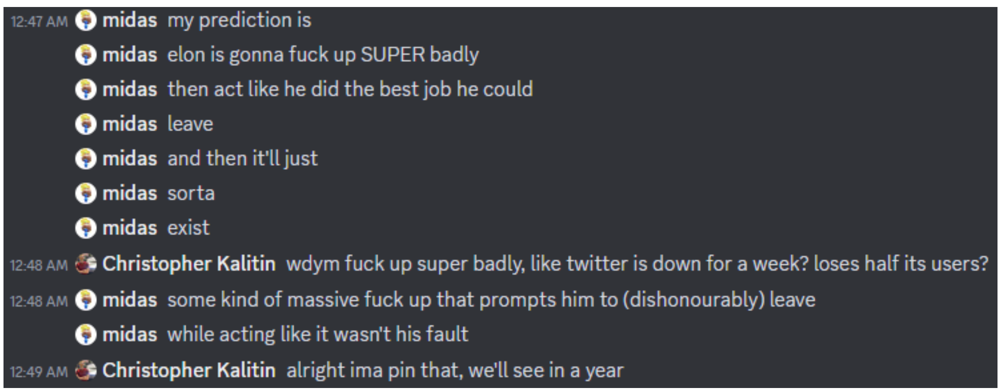
Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Only Make Bets You Can Win

Dec 17, 2023 • Christopher Kalitin





About one year ago on November 20 2022, I had a very unproductive and slightly useless conversation with a friend. We debated (argued) about the future of Twitter. With my obviously immensely high-rung and intelligent thinking I came to the conclusion that Twitter would have more features, revenue, and users than at the time. He believed there would be "some kind of massive fuck up that prompts him [E] to (dishonourably) leave while acting like it wasn't his fault."

"Unproductive discussions always feel like talking to a brick wall / low-rung thinker, even when you're the low-rung thinker." - Tim Urban, *What's our Problem"* (paraphrased). Even with this context, it's clear to me I was the high-rung thinking. A dangerous thing to say, but only one of us is writing a blog post dissecting this bet.

This conversation was very useful to dissect and learn about human nature. This blog post is an exercise in the proper way to approach past experiences. Dissect them for all the insights you can gain. Obsessing and feeling anxious about past events in your life is useless, but extracting insights is immensely useful. It's annoying how few people do this. The most insightful moment of my life was my first gf breaking up with me. Why does no one else dissect their past relationships!

Who Won?

His win condition:
"some kind of massive fuck up that prompts him [E] to (dishonourably) leave while acting like it wasn't his fault."

1. Elon has not dishonourably left and has admitted mistakes (Eg. Portland Datacenter)
2. No massive fuck up that prompted him to leave

My win condition:
More features, revenue, and users

1. There are obviously more features (X premium, Subscriptions, Monetization, Livestreaming, View count, Community Notes expansion, Hiring, Calling, etc.)
2. Decrease in revenue with advertisers leaving (They don't believe in free speech) and a shift to other revenue sources.
3. Some debate based on different metrics if users are down or up. Daily VS MDAU Vs User minutes. For these purposes, I'll go with users being down.

He was 0% right, I was 33% right.
I WIN!

This is far too close a result for me to ever make a bet like this ever again. This illustrates the need for more concrete metrics when making bets (a single win/lose condition, not separate) and that when you bet you must put a lot of thought into your beliefs.

I was 33% right in my predictions. Good enough to win, but not to feel good about myself. 67% wrong = mostly wrong!

This stems from a misunderstanding of where Elon would take the platform. I had no clue advertisers would leave to the extent they did and the extent of the battle for free speech. If I had put more thought into it, this could've been explicitly predicted - although it would be very difficult.

I've made several bets (for low dollar amounts), and these have all been useful exercises. Making bets is an excellent way to check if you really believe something. Can you sleep at night after making that bet? If you're wrong you lose money.

Future of X

More predictions, this time with more thought put into them.

It is clear Elon wants to move X away from advertising as the primary source of revenue. I doubt advertisers will ever be completely rid of the platform, but other sources of revenue will become a higher percentage of total revenue.

The primary motivation to acquire Twitter and create X appears to be the facilitation of free speech. Advertisers in the way? Tell them to go fuck themselves.
“If someone is going to try to blackmail me with advertising, blackmail me with money? Go Fuck Yourself.” - Balls of Steel

I hope X will gain more users as this shows the slow death of legacy media and the rise of free speech. On a long time horizon, I see this as likely, with the (temporarily) increasing human population if nothing else.

Feature development will obviously continue and the company will only employ the best available hard-working employees. As they should, who wants to work with lazy people?

1. X will slowly move away from advertising as a primary source of revenue.
2. User count will increase from today in the long term.
3. More features will be added.
4. X will become profitable within the next 5 years.
5. Free speech will be protected above all else.

Anyone wanna bet on these terms?

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Art Is Dead?

Oct 21, 2023 • Christopher Kalitin

Bo Burnham wrote Art Is Dead in 2010, well before 'Netflix-special' Bo Burnham existed.

The entire narrative of the song is incorrect.

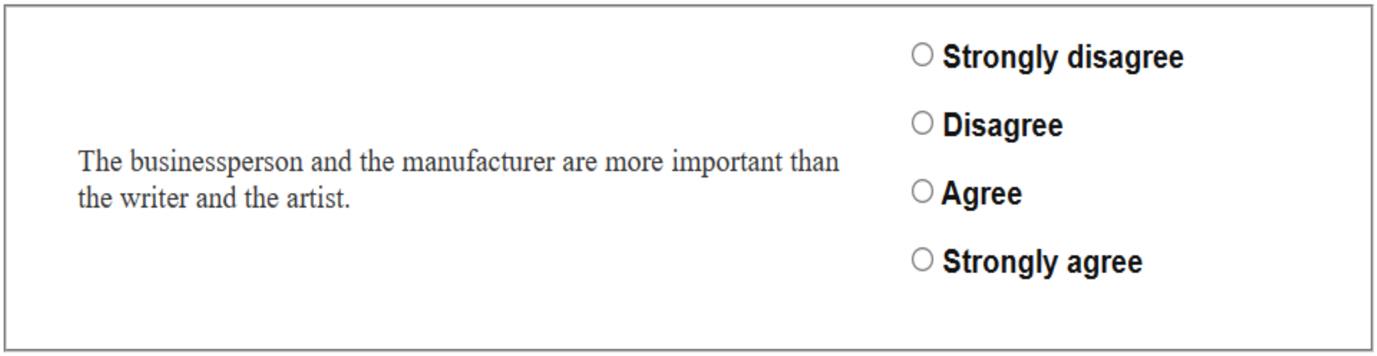
The lyrics of the song describe regular working people who pay money and attention to artists. The thesis is that artists - teenage Bo Burnham particularly - do not deserve this attention and are inherently phoney.

This is obviously not a fully formed thought or thesis about the world. Instead, it is the opinion of a teenager. Almost like my blog posts!

I must be psychotic
I must be demented
To think that I'm worthy
Of all this attention
Of all of this money, you worked really hard for
I slept in late while you worked at the drugstore

Fundamentally the trouble is teenage Bo Burnham does not feel it is fair that he can do fairly little work for a vast amount of money while many people are in the opposite situation. He does not understand the free market. You monetize the value you provide, whether it is selling drugs, growing crops, or making people laugh.

This is the juxtaposition in modern society between the artist and the labourer. This isn't a particularly perfect framework, but some people think it's useful. Both the artist and labourer are fundamental to society. Both are absolutely necessary in different ways, as with men and women.



Political Compass Test

The lyrics are fundamentally on the side of the business person and manufacturer. In other words, teenage Bo Burnham is obviously conservative. This appears to be attributable to a teenager not knowing who he truly is. This is what leads to the 'I'm depressed but it's funny' nature of Inside and millenials joking about themselves not being fully developed adults.

Entertainers like to seem complicated
But we're not complicated
I can explain it pretty easily
Have you ever been to a birthday party for children?
And one of the children
Won't stop screaming
Cause he's just a little attention attractor

When he grows up
To be a comic or actor
He'll be rewarded
…

These lyrics are an interesting take on how comedians come to be. Seems mostly accurate, I'll save the blog post on how we develop our personalities for the future.

Art is dead
So people think you're funny
How do you get those people's money?
Said art is dead
We're rolling in dough
While Carlin rolls in his grave

This is fundamentally where I disagree with the premise of the song. Yes, you're incredibly funny. Yes, you should make fuck tons of money for that. Maybe the comedy isn't as pure, but this allows for more comics. I love free markets.

The Mr Beast business model is the perfect anti-thesis to these lyrics:

1. Make a video about giving away money
2. Through the proceeds from this video give away even more money in your next video
3. Repeat until you possess and have given away infinite money

This is the most amazing application of monetizing attention on the internet. An infinite money machine centred around amazing content and giving away money. “We live in an attention economy.” - Andrew Tate

We should aspire to have more Mr Beasts in the world. People who love their work and do amazing things with their skills. I don’t want more slightly depressed and aimless ‘adults’ in the world.

Regardless, as with many other songs, I will continue to listen to it because it sounds nice. They’re really gonna make me vote for joe biden.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Waste Your Vote! Do What's Funniest!

Oct 17, 2023 • Christopher Kalitin

Some things I say are sarcasm, but all of my sarcasm is based in truth. This is evident in my first post. I will let you figure out where the line is.

If it is not already clear to you: your vote does not matter.

I will use data from my riding in the previous BC general election to illustrate this point.

In the past 3 elections in the Burnaby-New West riding the NDP has won with ~45% of the vote. This data only goes back 3 elections and there have been no real differences aside from independent candidates and minor parties joining and leaving the elections over the years.

I will bet $1000000 that the winner of the next election will be the NDP candidate. There is almost nothing I or any of you reading this can do to change this outcome. If anyone would like to take my bet, email me: Christopher.Kalitin@gmail.com

There are countless ridings just like this one. If you are not in a swing riding your vote doesn't matter. The same goes for states as well.

In the last general election in Canada, there were 17 million votes cast. 62.2% voter turnout. Rejected ballots have been increasing in the last 3 elections in my riding. There's hope!

When hearing this 17M number you may come to the conclusion that 17 million people controlled the result of the election. This is not the case. As I laid out above, only swing ridings matter. California's gonna vote blue, Texas is gonna vote red.

What is the real number of voters that have a disproportionate impact on the election? According to this CBC article there are 60 swing ridings out 388. 15%. 17M*0.15=2.55M.

If you live in a swing riding, take your vote seriously. If not, have fun with it!

In my mind, there are only two valid reasons to vote if you don't live in a swing riding/state.

First, posterity. You can prove your previous thoughts through your voting history or allow random teenagers to analyse the data years later. Statistics is a great reason to vote.

Second, practice. If you take voting seriously you may get better at it. Sadly the vast majority of people don't do rigorous research before voting - this goes for every side of the political spectrum, liberals who talk about "research" while not changing their thoughts are not free of blame. I will never again speak of the left and right is such an unacademic way. Tim Urban speaks in a very academic way about such problems in his book. Read it.

I see the second reason as less valid than the first. I have only seen one person ever talk about voting the way people should discuss it, Tim Urban. Read his book!

I've covered people who have valid reasons to vote. Now, for the majority of you reading this.

DON'T VOTE!

By voting you are supporting representative democracy. Fuck representative democracy. It's horribly underoptimized. If we decrease voter turnout enough, something must happen to overthrow representative democracy and replace it with something better!

Centralizing power appears to have often been a bad thing throughout history. Representative democracy centralizes power with politicians. Better than aristocracies, but still not ideal. "Capitalism is the worst economic system, except for all the others."

I've come to a conclusion given these two insights: (i) your vote doesn't matter and (ii) supporting representative democracy is bad. My conclusion is you shouldn't vote, or at the very least you should waste your vote.

For voters in Canada, there is an obvious option: Vote for the Green Party!
The Green Party will never win! They get under 5% of the vote! Might as well vote for them for fun! I support the turtles!
Or even better: Vote for Quebec to succeed! Give it back to France!

Humor is the only reason to vote!

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

You Will Get 80 Trillion Dollars If You Read This Closely

Oct 11, 2023 • Christopher Kalitin

It's true that an author will do just about anything to keep your attention, but I am serious about the title of this post.

You must apply the Law of Accelerating Returns to your investment philosophy.

If you bought Tesla stock 10 years ago and held you'd be up 21x
If you bought Apple stock 10 years ago and held you'd be up 9x
If you bought Amazon stock 10 years ago and held you'd be up 7x

With this data is it obvious that predicting the next companies to 10x is the best investment strategy. 10% per year in the S&P 500 is fine, 2.6x over 10 years. However, a 25% CAGR with companies that 10x every decade will make you a billionaire.

Now the question has been reframed.
It is no longer: "How do I invest." This is obviously a question for dumb people.
Now the question is: "Which industries will undergo exponential growth over the next 10+ years and which companies will be the winners in those industries."

It is important to start with industries, not companies. Companies are individual entities so they are difficult to predict individually. Entire industries are much larger and thus easier to predict.

The Law of Accelerating Returns and S-curves can be applied to any segment as a method of understanding if that segment will undergo exponential growth.

The only method I've found for predicting which companies will be the winners in a segment is to extrapolate the future progress of current companies. I have made models of Tesla, SpaceX, RocketLab, and NVIDIA. As opposed to industries, these models require much more thought and effort. It is much more difficult to predict the future of a company than the future of an industry, there is a reason Kurzweil predicted industries and not companies.

I am making many predictions now so that in 10 years I will understand if I was correct or not.

That is the purpose of this post.

These are my predictions for the future:

1. EV adoption will follow an S-curve and will be greater than 80% of new car sales by 2030.
2. More than 50% of total kilometres driven will be autonomous by 2033.
3. The growth of renewable energy will follow an S-curve.
4. In 2050, solar will be the largest source of energy on Earth. Barring any fusion breakthroughs.
5. Space Applications (eg. Starlink) will grow exponentially long into the future.
6. AI Robot's takeover of labour will follow an S-curve. This will be the largest economic paradigm shift in human history.

A note on fusion: If there is a fusion breakthrough that allows for cheaper energy generation than future solar, its adoption will follow an S-Curve. Currently the S-curves of renewable energy are much clearer than that of fusion, as fusion is an R&D project. In the future, it may make sense to invest more in fusion but currently it is high risk and difficult to invest in.

With the predictions I made above, I can now make predictions about the future of industries and companies.

1. There will be a paradigm shift in the automotive industry away from personal ownership of ICE vehicles to Autonomous Robotaxi EVs.
2. The energy industry will be dominated by solar and wind.
3. Space Applications will grow exponentially.
4. The use cases of AI will grow exponentially.

Which companies are poised to be the winners in these industries?

1. Tesla - Leader in EVs, Autonomous Driving, Solar Energy Generation, and Energy Storage
2. SpaceX - Leader in Space Launch and Space Applications
3. Rocket Lab - Second Place in Space Launch and Space Applications Hardware
4. NVIDIA - Leader in AI Training Hardware and Software, although hopefully not for long

You don't need to invest in the industry leader. If you want to be safe, go ahead and buy to S&P 500 too.

To illustrate this point, Rocket Lab is starting from a much lower base than SpaceX. $2B vs $150B market caps. Easier to compound at a lower base. I'd bet that Rocket Lab's launch business will compound more than SpaceX's in the next 10 years. 4x vs. 7x by 2031 in my Models.

Comma AI is worth ~$100M, Tesla is worth ~$800B. Comma will hopefully never "sell out to the bankers" so good luck investing in them.

Ease of investment is also a concern. Rocket Lab and Tesla are public.

Short-term investing is for idiots, obviously.

Short-term investing is predicting the stock market.
Long-term investing is predicting the company and the segment.

It is possible to successfully invest short term. But it is much more difficult and less interesting. Do you really want to worry about what the FED will do to interest rates next month? I'd rather model the future of humanity.

To further illustrate this, consider the CAGR/time ratio. What is the point of day trading 10hr/day when you can get similar returns in the S&P 500 for 1s/day.

The economy is growing exponentially, this is what allows for seemingly impossible predictions.

In 1980, the company will the highest profit was Exxon Mobil with a profit of 4.2 billion dollars.
In 2000, it was General Motors with a profit of 6 billion dollars.
In 2022, it was Saudi Aramco at 300 billion dollars. Apple in second place with $122B.

In 2000, Microsoft was the most valuable company at ~400 billion dollars.
Now, Apple is worth 2.8 Trillion dollars.

This is why you will get 80 Trillion dollars. Extrapolate these exponential trends into the future.

If the investment philosophy laid out above is correct, I will be a billionaire.

One million is only 3 10x's away from one billion. This will take 3 decades from the point where I have one million dollars to become a billionaire. That is a long time, but it is achievable.

you - pronoun
Used to refer to the person or ***people*** that the speaker is addressing.

Kurzweil began the chapter 'The Singularity as Economic Imperative' in The Singularity is Near the same way I began this post. "You will get 80 Trillion dollars just by reading this section and understanding what it says"

You collectively will all get 80+ trillion dollars if you understand this post. Although, depending on how far you look, this number could be orders of magnitude higher.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

NASA's fucked - Here's my vision

Oct 1, 2023 • Christopher Kalitin

How’d This Happen?

In researching NASA I have come to the obvious conclusion that it is a horribly bureaucratic and political organizaiton. I haven’t seen anyone ardently dispute this point. As I am writing this post I see the Moon rising over the horizon. NASA has failed. We haven’t colonized it yet.

“NASA hasn’t made any good rockets since the 60’s” - Geohot on a stream.

How do a few kilograms of Mars rocks cost $10B? How does the SLS cost >$40B before its first flight?

There are two core explanations:

1. Cost-Plus contracting
2. Politics

Cost-Plus contracting gained a lot of popularity during the Second World War. The government needed to build a lot of stuff quickly and cost-plus was the best way to do it.

This is the Cost-Plus process: Your company gives an estimated price for a contract. If you need more money, oh well, we’re at war, you can have it. No one would dare overspend money if your coworker’s son would die. [Update Dec 7 2024]

No one dies if SLS is delayed 5 years. So, the contractors can delay and get more money. Incentives are not aligned.

The other explanation is politics.

The Senate Launch System (SLS) and Artemis Program are prime examples of NASA’s inefficiencies.

They were born out of the constellation program. Ozan Bellik’s blog posts have a lot of good info on this.

The Constellation Program's plan to land on the moon involved two launches that docked in Low Earth Orbit. (1) An Ares 5 launches a lander and a TLI stage. (2) An Ares 1 launches an Orion capsule into orbit which docks to the prior launch's payload.

When Constellation was rightly cancelled by the Obama Administration in favour of Commerical Space (lots of info about this in Lori Garver's book) the Senators complained about all the government-funded jobs that would disappear from their districts with the end of the Shuttle hardware. So, the Senators came up with the Senate Launch System (SLS).

The SLS is an underoptimized rocket for its task: Landing on the Moon. A single-launch lunar landing is absolutely possible with it's 95-Ton LEO Capacity. However, the mass of Orion and its low-deltaV service module prevent this. Under the Constellation Program, the Altair lander would conduct the Lunar Orbit insertion burn. The Orion CSM is not designed to get into Lunar Orbit on its own.

The Constellation Architecture is not inherently flawed. But it takes NASA in the wrong direction, as I will explain later. In contrast, the Artemis Architecture takes a couple of parts of Constellation and a few parts of Commerical Space and joins them together in an underoptimized Lunar Program.

At least we get Reusable Lunar Landers out of Artemis.

My predictions for Artemis are the following:

1. The SLS will be cancelled when there is a suitable replacement.
2. These replacement(s) will be derived from the reusable landers.
3. The later Artemis missions will become more commercialized.

Artemis is far from the only example of NASA's deep inefficiencies.

I want to be clear. NASA is a net positive for the world. But it is horribly inefficient. Government agencies often are.

There are many examples of NASA impeding progress in space. Lori Garver's book Escaping Gravity has many examples of this. As Deputy Administrator of NASA, she was in a position to see this.

Who was against the commercialization of NASA? Who stands to lose if NASA becomes more efficient? The contractors and senators.

These are the people who spoke out against the COTS, CRS, and Commercial Crew contracts. It is debatable whether they truly believed the commercial companies would fail or if they were just trying to keep their funding [See Update Dec 13 2023]. The true answer obviously lies somewhere in the middle. However, with the last decade of progress in commercial space, it is clear that the commercial companies were the right choice.

It is a testament to the senators and contractors power that they could overcome a directive from the President of the United States. President Obama was in favour of the Commercialization of NASA.

With the recent OIG report on the cost overruns of the Mars Sample Return program (MSR), I saw a post from a former JPL employee that is very relevant to this discussion. Backup image in case the original is deleted.

It shows how SpaceX pursued Red Dragon as a Mars Lander. This would be a more efficient platform for conducting science on the surface of Mars. There was intense pressure to not give funding to this program. From the post, it is not entirely clear where this pressure came from or the exact motive. But it is clear that the pressure was enough for SpaceX to cancel Red Dragon and shift over to Starship.

JPL currently conducts all the flagship missions to Mars. They are very good at getting small payloads to the surface of Mars. But they are not good at decreasing the cost - MSR is $10B! Red Dragon could have decreased the cost per ton to the surface of Mars by more than an order of magnitude.

Who stands to lose from this? JPL and contractors.

This shows the common theme with NASA projects: entrenched interests that are not aligned with the goals of NASA.

If NASA could proportion funds to maximize science and exploration without regard to outside interests, it would be a much more efficient organization.

One of the factors that prevent this is how the US Federal Budget is allocated.

NASA cannot simply shift funding from one program to another. They must spend the money on the SLS that has already been allocated to it. Congress is not an efficient means of allocating funds. Mini-dictatorships are more efficient, also known as private companies.

Vivek Ramaswamy has spoken about how to fix this. When constructing the Federal budget we should work from the ground up to determine what resources are truly needed. Instead, we work from the top down. This means that the budget is not optimized for the best use of resources.

The system that created NASA’s inefficiencies wasn’t completely an accident. NASA’s distributed nature was planned from the beginning.

Because NASA has centers in multiple congressional districts it has multiple Senators and Representatives that will fight for funding. This keeps NASA’s funding high and stable. But it also means that NASA is a jobs program.

I thought NASA was meant to explore space and do science. Not be a jobs program.

How to Fix NASA

What is the goal of NASA?

If you ask most people I hope they would say to do science and explore space.

There is a rhetoric that being a jobs program for people who work on out-of-date technology is a good thing. I disagree. Keeping people employed is good for those people and the economy, but not for science or exploring space.

It is very important to understand the goal of NASA: Science and Exploration. Otherwise, you get sucked into thinking NASA should be a jobs program.

Now that we have set the goal, we can use it as a framework to analyze NASA’s current programs and predict its future.

Show me a single example where a government agency has been more efficient than private companies. I can’t think of a single one off the top of my head.

NASA should be a customer, not a provider.
“One customer among many” - Jim Bridenstine, former NASA Administrator

Private companies have proven their ability to get to space cheaper and more efficiently than NASA. This is why for most payloads, NASA uses commercial launch providers. Private companies can also get people to space cheaper and safer than NASA could. This isn’t purely SpaceX’s doing, but they have been the most successful at it.

Is it too far of a stretch to assume that private companies will be able to build and operate space stations, rovers, bases, Moon landers, etc. better than NASA could?

Commercialization is inevitable either way. We might as well embrace it early.

The early voyages to North America were government-funded. How many are today? Almost none aside from the military. The early voyages to the Moon were government-funded. How many will be entirely government-funded in the future? None (ish).

I hope NASA never builds a moon base. They are bad at building big orange rockets, so it's not a stretch to assume a NASA moon base would be far over budget and behind schedule as well.

Commercial Space Stations are already in development and will fly in the next few years. They don't have a proven track record yet, but I am confident they will be successful in being cheaper and more efficient than the ISS.

If Space Stations are commercialized, why not Moon Bases? The technical challenges are not insurmountable for private companies.

Even with the commercialization of space, NASA still has a role to play in development programs. SpaceX would not be where it is today without NASA funding and collaborating with NASA on development. This is especially true for Crew Dragon.

In the next 10-20 years NASA will still need to be involved in private development programs. They will provide funding and expertise. This will allow for privately development programs that can be cheaper and more efficient than NASA could do on its own.

NASA is a stepping stone for commercialization.

The primary challenge in the future of completely commercializing space is the source of funding. How will companies make money if not from NASA? In-space manufacturing and mining offer a solution, but we will have to wait 10-50 years to see this play out.

What will NASA's role be in the commercialized future of space?

NASA will use technology and spacecraft developed by private companies to conduct science and exploration.

This is already happening with the CLPS program. NASA is paying private companies to develop Moon landers. NASA provides the experiments. The private companies provide the landers.

The Commericalization of space is already occuring as private companies are creating an economy in space. NASA should embrace this and use it to its advantage.

This is the first era of commercializing space. NASA provides funding, technical expertise, and the payload. In the next era, private space companies provide End-to-End solutions for missions.

The payloads of NASA missions will be commercialized. Currently, NASA is very much involved in the development of its own satellites and Mars landers. In the future, NASA will provide the mission requirements and private companies build the satellites and landers.

Commercialization has been the trend in all industries. It is inevitable that this will happen in space. It is only a matter of time.

Updates

[Update Dec 13 2023]
With Smarter Every Day's latest video, it seems much of the personal push from engineers is good hearted. It's only when the government gets involved that degeneracy ensues. Also, Destin makes the mistake of simplifing the architecture at the expense of the mission.

[Update Dec 5 2024]
Ozan Bellik summarized the insights in this post well by saying systems engineering and project management should be moved to private companies, while NASA mainly does R&D, data analysis, and mission planning.

[Update Dec 7 2024]
The fundamental issue with cost-plus contracting now is that contractors are not incentivized enough to move quickly. In World War 2, the contracting paradigm didn't need to include incentives to move quickly because it was implicit in the national priorities of the time (not losing the war). Now, contracting paradigms need to include urgency to complete projects on time because there isn't this external signal to move quickly.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Applying the Law of Accelerating Returns

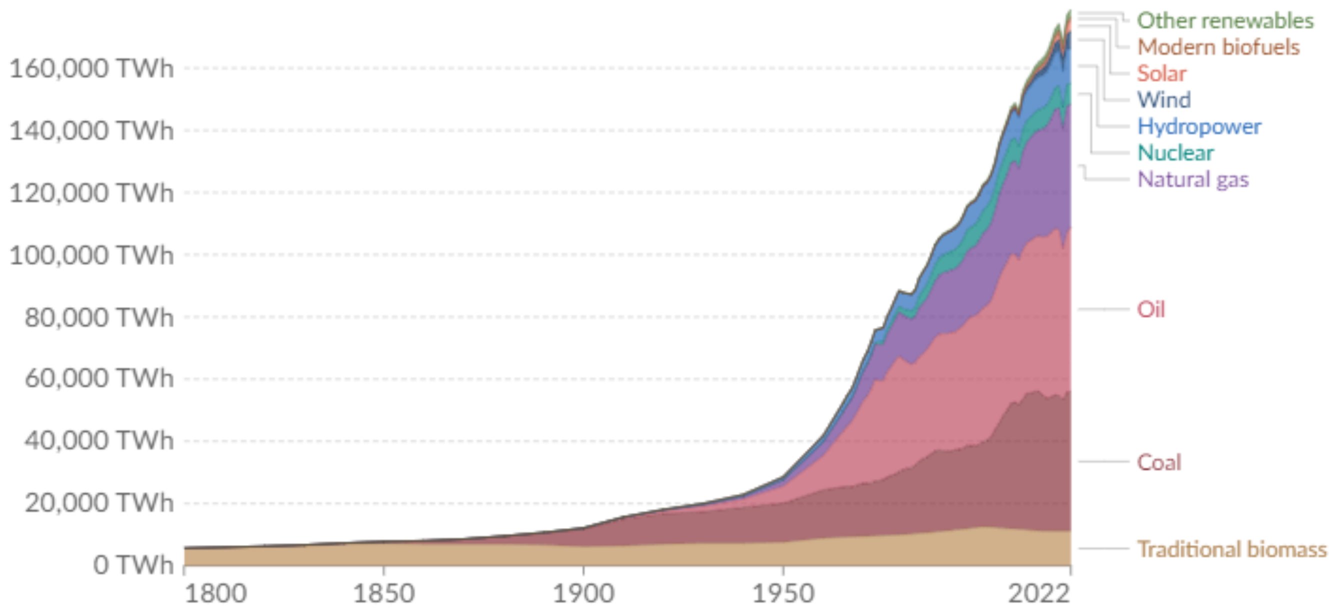
Sep 22, 2023 • Christopher Kalitin

I've read a few of George Hotz's blog posts. So I'm starting a blog. This is a first, late-night, attempt at a blog post.

In predicting any part of the future, I aim to be as accurate as possible. Emotions are the antithesis of this, emotions are not accurate. Predicting the future of humanity has nothing to do with my optimism or pessimism about it. I want to be right.

I do not solely aim to spend my life predicting what others build, but before you can build the future you have to know what to build. Kurzweil says it well, "As an inventor in the 1970s, I began to realise that my inventions needed to make sense in terms of the enabling technologies and market forces that would exist when the invention was introduced as that would be a very different one from the one in which the way was conceived."

The Law of Accelerating Returns is the most fundamental aspect of technology that everyone needs to understand. I have seen friends say "we won't have electric trucks for 20 years" or "AI cannot take over the world". These points fundamentally misunderstand the exponential growth of new paradigms, the S-curve. Furthermore, these points miss the exponentially increasing growth rate of humanity, see the energy chart. Humans are bad at understanding exponentials.



Source: Energy Institute Statistical Review of World Energy (2023); Vaclav Smil (2017)
OurWorldInData.org/energy • CC BY

Why is wood still used?

We need to use more energy. Energy consumption is closely linked to quality of life and life expectancy. Using more energy saves and improves lives.

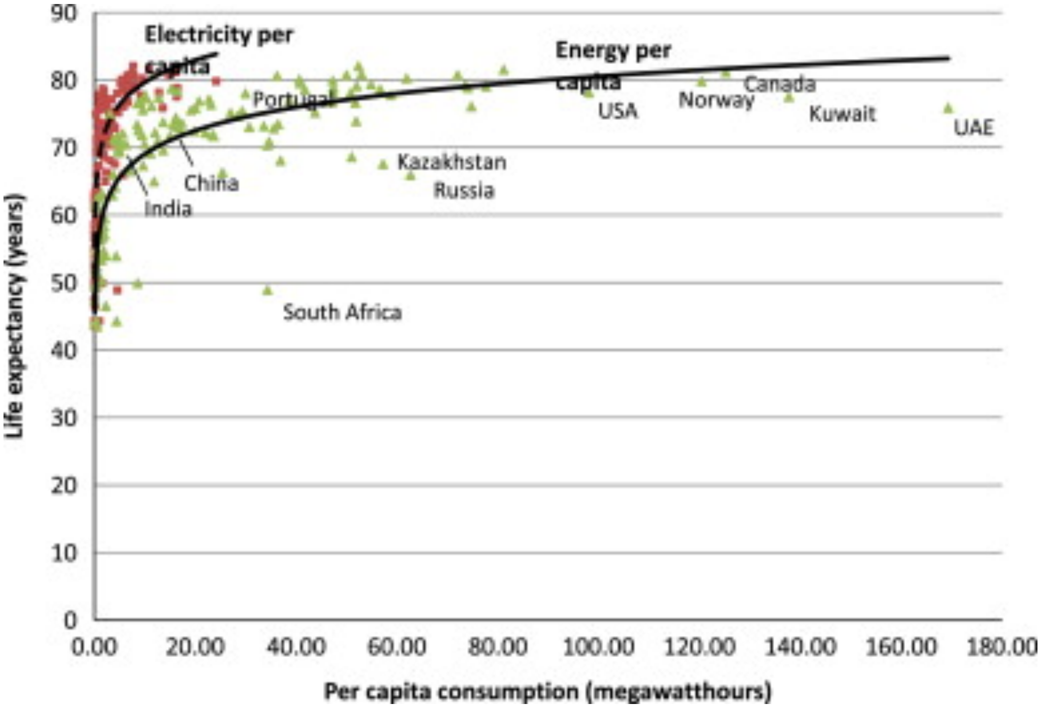
Furthermore, I believe advancing up the Kardashev Scale is a moral imperative for humanity. I do not see any other goal for humanity than understanding the universe and becoming a galactic civilization helps us do this. The more energy we have at our disposal, the more can be learned about the universe. Put differently, we must use energy to meet god.

The only way to progress on the Kardashev scale is to produce more energy. How will we incentivize this increase in energy production? Simple, increase demand for energy. Use Air Conditioning when you're not home, run all the TVs in your house all day, the lights too, keep your stoves running to burn more gas, and keep computers running at full capacity at all times. This is a moral good for the future of humanity.

It is a damn shame the chart of energy consumption hasn't grown faster. Let's fix that together by using as much energy as possible. Damn the efficiency, the Kardashev scale only measures production!

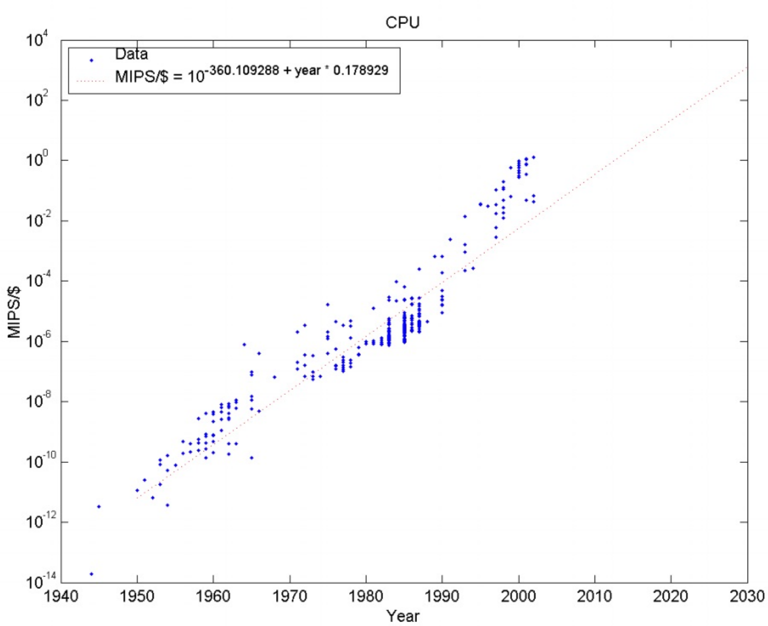
A note on climate change: I don't care if some Pacific island nations get flooded. The Kardashev scale is more important. Also, I will personally kill every endangered species within 100km of Starbase if it means Starship launches 1 month sooner.

Climate Change will not kill us. The rapidly advancing paradigms of clean energy will take over the world and save us from climate change for a simple reason: they're cheaper. Installing new solar is cheaper than coal. EVs are cheaper than ICE. My prediction for EV new car market share is ~50% in 2026 and 90% in 2030. S-curves are easy to project into the future.



The long tail of life expectancy is not ideal. Hopefully, we'll start living longer soon.

I have seen no particularly eloquent and condensed definitions of the Law of Accelerating Returns in my Google search of the definition. This is my definition: The growth rate of technology is exponentially increasing. S-curves are getting faster. Our kids will grow up in a drastically different world than our own. Better have them quickly!



MIPS per dollar, Log Scale. Moore's law isn't dead. Strangely, it hasn't accelerated.

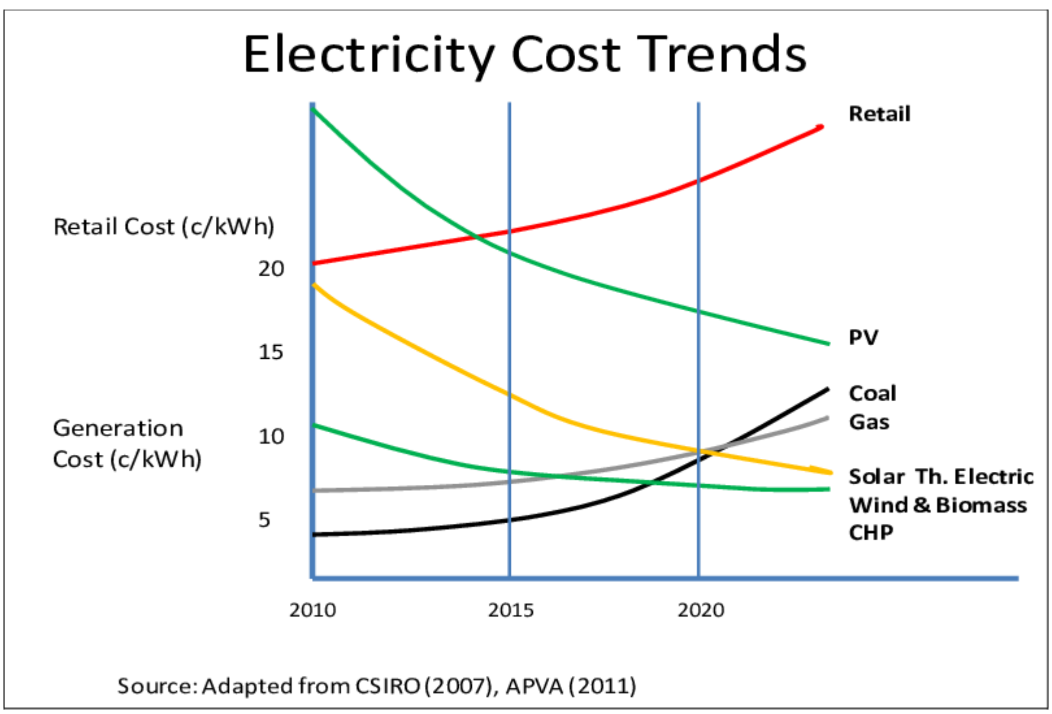
The accelerating growth of humanity and the accelerating pace of S-curves are certainties in my mind. Fundamentally, for a new and improved paradigm to take hold the legacy paradigm must come to an end. Energy is the prime example.

The stagnation of the cost of energy is described in The Singularity is Near. Coal, Oil, and Natural Gas produce most of our energy and they are evolved technologies. Their growth rates are so slow that the increasing fixed costs cause the price of energy to increase. Or maybe the companies just like money, Capitalism isn't always perfect.

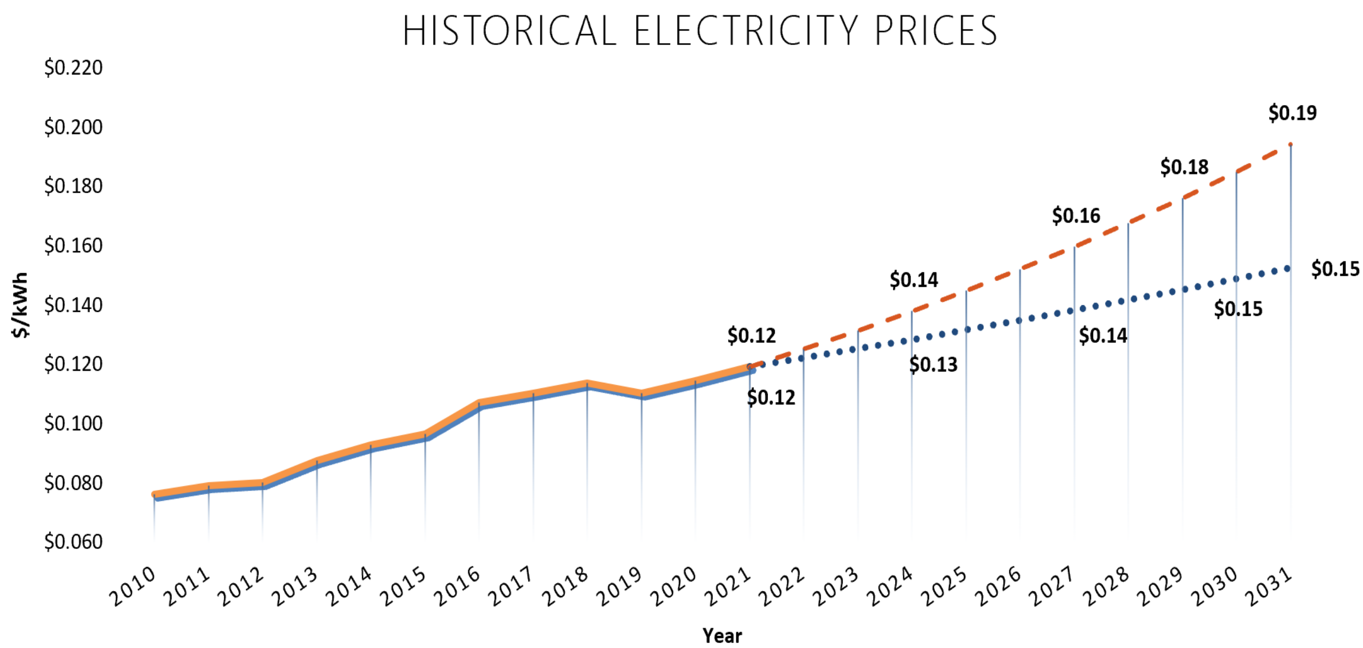
Renewable energy fixes this. Solar and wind energy generation are on declining cost curves. We are currently at the inflexion point where installing new solar is becoming cheaper than fossil-fuel-based alternatives. Soon, Solar will take over the world. In some geographics wind or hydro will be more effective. But even in BC - where BC Hydro produces the vast majority of our energy - installing solar panels on your home is a profitable long-term financial decision.

Other stagnant fields ripe for disruption are Internal Combustion Engine Vehicles, Healthcare, Education, Rural Telecommunications, Government, centrally controlled currency, etc.

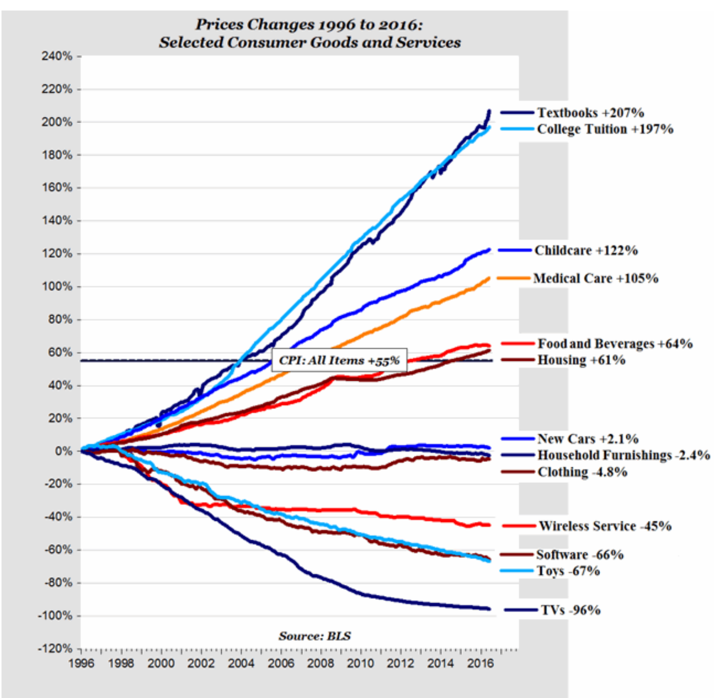
Crypto is excellent for regulatory arbitrage and fighting authoritarianism. Take the power of printing money away from the government and give it to no one.



Solar ftw, retail cost explains BC Hydro graph below.



BC Hydro is increasing energy prices. We will put them out of business with solar. Only evil people increase the price of energy.



Anything above the line should be disrupted. Housing too. Buy Tesla stock, not real estate.

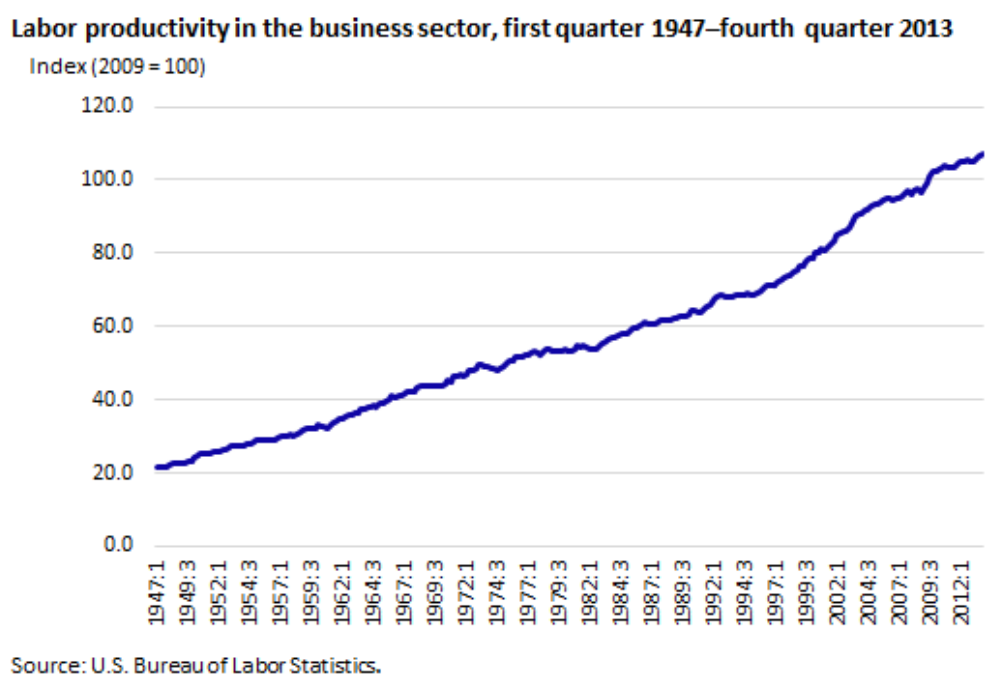
The question of how exactly AI will impact our lives in 50 years is a very difficult one. The growth of AI is too fast to predict its impacts long into the future. Luckily I don't need to know the exact use cases of AI in 50 years to work on it today.

The exponential growth of AI should strike fear into everyone's hearts. ChatGPT did this for a while, then it became normal.

A very intelligent scholar on Twitter explained the recent progress of AI well, “In 2024, AI advancements will be more numerous and happen faster than in 2023.” This will be true for every year long into the future barring any humanity-scale catastrophe.

The current paradigm of using AI systems as tools is fairly easy to broadly predict. AI tools will replace large parts of many workflows and allow for more output per hour in applicable fields.

Workers will become more efficient. This is usually put into one of two categories: companies will need fewer workers and fire those not needed or the productivity increase will allow for more work to be done. One of these scenarios increases the size of the economy, the other does not. I lean towards the scenario that increases the economy being more true than the alternative.



Increasing worker efficiency is not a new thing.

There is agreement between George Hotz and Elon Musk that Kurzweil is right in his 2029 prediction of a functional simulation of the human brain. The Singularity. I will use this 2029 date.

This means in 2029 silicon-based intelligence will be able to do the work of biological intelligence. It is not a perfect, overnight transition, but that is precise enough for my purposes.

What happens after this?

For many paradigms, it seems impossible to accurately predict AI’s impact. So, I won’t.

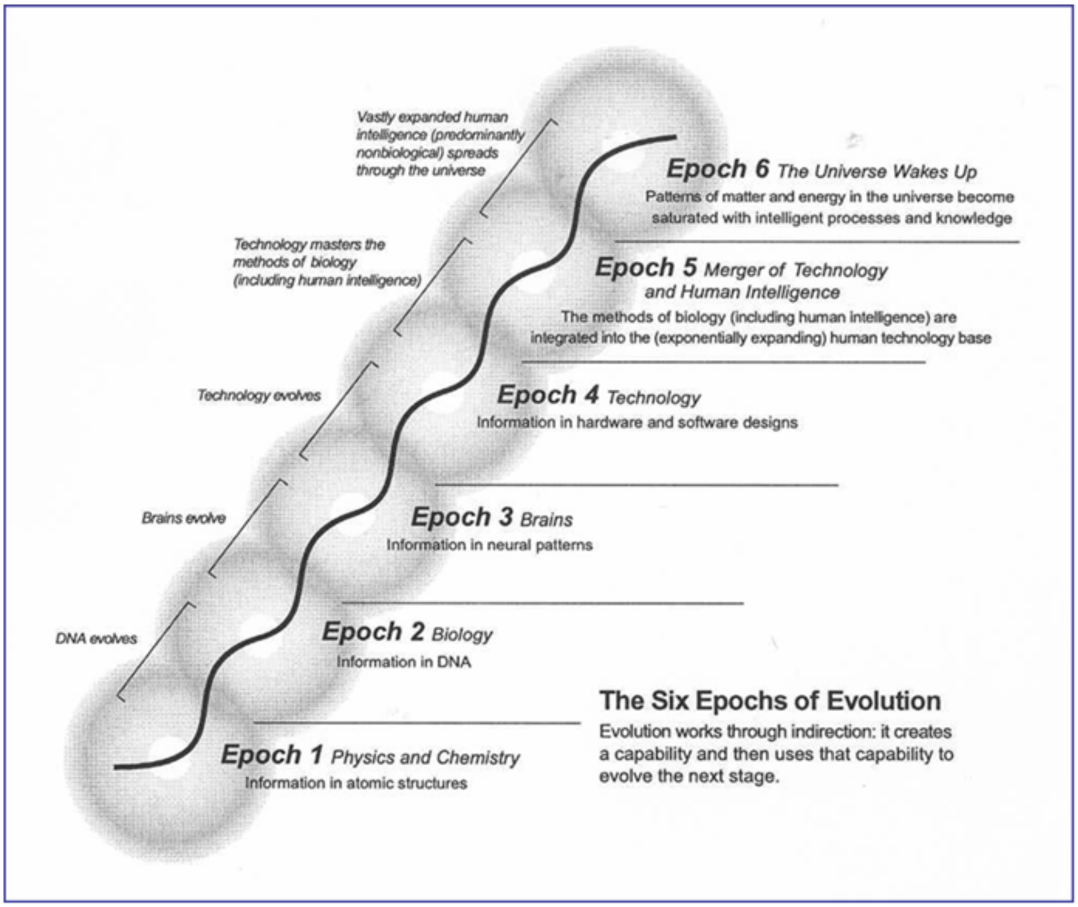
It is also impossible to accurately predict the colonisation of the universe. But the Kardashev scale gives us a framework and the speed of light sets an upper limit.

What is the framework for predicting the rise of AI? I believe it is intelligence.

AI will greatly increase the amount of intelligence in our part of the universe. This will make progress on the tech tree much faster. This will allow us to meet God - whatever created the universe - faster.

This is the extent of my predictions on the future of AI. At the moment, I do not see any other accurately predictable outcomes.

If this is too abstract for you, read The Final Question by Issac Asimov.



Epoch 6 is the important one for meeting God.

Aside from a Yudkowsky-style Authoritarian halt to AI progress or a Butlerian Jihad I see no way to stop AI. The advancement of technology has never been stopped in the history of humanity, it has only been briefly delayed.

Slowing down AI progress is certainly a safer approach than progressing at full speed. 6-month moratoriums are not enough for this. Regulation helps to prevent our death at the hands of AI, but I am not a fan of increasing the power of the government. So, this is again an unsatisfactory solution.

I currently have no concrete answer as to how to prevent the potential negatives of AI. The right people must be in charge. ClosedAI are not the right people.

To conclude I will summarise my beliefs/predictions.

1. The goal of humanity should be to uncover what created the universe. Meet God.
2. Advancing up the Kardashev scale helps us meet god.
3. If The Law of Accelerating Returns is understood, it is trivial to predict the growth of some paradigms. Clean Energy, EVs, etc.
4. The advancement of technology cannot be stopped.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

Stoke Nova Cost Model - Will it Survive?

Jun 25, 1990 • Christopher Kalitin

Tim 30% hardware cost post: https://x.com/Tim_X94/status/1802451587073225187

Whiteboard notes:

- Cadence impact
- Market addressability given mass (Eager space chart)
- 30% hardware cost? What?
- Model net company costs

Vulcan reuse numbers: https://x.com/deltaIV9250/status/1878461890923217295

Ozan, Reuse numbers for external parts like Upper stage, fairings, boosters, etc: https://x.com/BellikOzan/status/1878602456537682227

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

The Bare Minimum Engineer

Jul 20, 1969 • Christopher Kalitin

Longer title: How To Become The Bare Minimum Engineer

This is partly a play on Casey Handmer's The Well-Rounded Engineer.

Slightly a continuation of How To Teach Engineering, though that was mostly just yapping.

Potential Structure:

1. Get on the Positive Slope Side of the J Curve
2. How To Become The Bear Minimum Engineer
3. How Not To Become The Bare Minimum Engineer

The Bare Minimum Engineer

1. Play on Handmer's The Well-Rounded Engineer
2. The MVP of an engineer, providing value
3. How to be a net positive in organizations

How to actually learn things and why university is inefficeint

1. Just take the conversation with Prashant and formalize it
2. Work from first principles

- Maybe note that I'm just a student? Open to iteration? Or just go for it? Just go for it.
- Get Mischa's feedback?

1. The Well-Rounded Engineer is about how to maximize your productive output as an engineer
2. The Bare Minimum Engineer is about how to get to the positive side of the J curve

3. As an engineering student, or young person in general, or new entrant into an engineering organization, you are a net negative
4. You need to acquire knowledge and skills to start becoming productive, and then pay back your debt to the organization (UBC Solar)

5. Be extremely curious, the side effects/fundamental reasons for this are extremely positive and increase your net lifetime productive output (increased salary too in engineering)

phenolphthalein

Prashant Conversation Notes:

- - - Hardest I've ever worked and not the most productive I've ever been
- - - You need to understand why you're studying something and what the goal is to know what information is important (Falcon 1 example toughness)
- - - Self motivation is extremely important. University's forcing function is extrinsic
- English class is low signal to noise ratio

- - - Projects that are useless in and of themselves are demotivating
- - - Complete focus = max productivity. Context switching is the mind killer (Joe Justice with Dr. Know It All Podcast)
- - - Feedback from people more intelligent than you is extremely important
- - - Classes mostly teach computations (maybe this changes in upper years) + tested on computations (MATH 120)
- - - Not incentivized to learn things from first principles
- - - Entry is based on grades
- - - First Principles give the side effect of being successful academically (mostly)
- - - No clue how to test. Do you even test? You have the real world, job interviews and portfolios are your test
- - - Back to the original point, you need to be able to integrate feedback
- - - First Principles and Projects show you why the things you're learning are important!
- - - High iteration rate is important and motivating, in contrast to fax projects
- - - Anecdotes? Nah fuck that I'm right. I am my father's son. Symptom of extreme self belief

Prashant conversation:

[7:24 PM]Christopher Kalitin: The paradigm of work in university is very inefficient [7:24 PM]Christopher Kalitin: This is simultaneously the hardest I've ever worked and not the most productive I've ever been [7:24 PM]Christopher Kalitin: In about the last year of online school I figured out the proper schedule to maximize productivity, sadly now this can't be utilized

- - - Hardest I've ever worked and not the most productive I've ever been

[7:27 PM]Christopher Kalitin: We're studying the fundamentals without knowing why, which greatly decreases efficiency

You need to know why you’re learning something to properly learn it

Like the transformer architecture, you need to weigh certain facts more highly than others - if you don’t do this weighting you’re far less efficient

The only way to figure out what knowledge to put more effort into learning than others (weights) is to actually do something, to build, this is why design teams are important

- - - You need to understand why you’re studying something and what the goal is to know what information is important (Falcon 1 example toughness)

[7:30 PM]Christopher Kalitin: You have to be self motivated to do it, this is the problem with university, nearly everything you do is externally motivated “oh no I need to study for this test or I’ll get a bad grade”

- - - Self motivation is extremely important. University’s forcing function is extrinsic
- English class is low signal to noise ratio

[7:48 PM]Pieman: It’s more about the practice and learning how to form a significant and debatable claim on a unresolved scientific issue [7:49 PM]Christopher Kalitin: Learning on projects that are useless in and of themselves - you need to write real research to know how to write research papers!

- - - Projects that are useless in and of themselves are demotivating

[8:55 PM]Christopher Kalitin: You’ll get good at writing papers when its all you think about - this doesn’t happen in school

- - - Complete focus = max productivity. Context switching is the mind killer (Joe Justice with Dr. Know It All Podcast)

[8:56 PM]Pieman: The willingness to wrote papers is all your own [8:57 PM]Christopher Kalitin: in university its extrinsic, not intrinsic

[8:57 PM]Pieman: Right you could have great ideas, but presenting them on paper is something that requires alot of practice [8:57 PM]Pieman: It requires the feedback of people that are more educated than yourself

- - - Feedback from people more intelligent than you is extremely important

[8:59 PM]Christopher Kalitin: In chem & physics this is especially clear because all you’re tested on and learning is how to solve equations - and if you have a good teacher many some first principle understanding

On projects, all that matters is first principle understanding of the systems you're working on, do chemists actually solve stochiometry problems by hand? This is extrememly clear on ubc solar

- ○
    - ▪ Classes mostly teach computations (maybe this changes in upper years) + tested on computations (MATH 120)

[9:04 PM]Christopher Kalitin: Exactly I completely agree

Now the more fundamental issue is that the incentive in university is not to learn things from first principles, it's to get a good grade on the test

I have a few friends like this and it's immensely sad and unproductive

Like you're saying, its a personal responsbility to try to understand things from first principles - personal responsbility, not academic responsbility

Your academic responsibility is to get a good grade on the test

- ○
    - ▪ Not incentivized to learn things from first principles

[9:05 PM]Pieman: If you see uni through only the lense of the goal of grades I doubt your gonna go far [9:06 PM]Christopher Kalitin: Thats the way the university sees it

Your entry into the university is mainly by grades You getting into the program you want is mainly on grades Etc.

- ○
    - ▪ Entry is based on grades

[9:06 PM]Christopher Kalitin: There is the side effect of understanding things from first principles that you'll probably do well on the tests - but this is not a direct effect, only a side effect

- ○
    - ▪ First Principles give the side effect of being successful academically (mostly)

[9:07 PM]Christopher Kalitin: Sure, but they're optimised for testing you on quantitative computational knowledge, not testing you on your first principles understanding of various systems, which is what actually matters on technical projects

[9:07 PM]Pieman: How would you propose unis test us then [9:08 PM]Christopher Kalitin: No clue yet! for now I've only identified the massive problem

- ○
    - ▪ No clue how to test. Do you even test? You have the real world, job interviews and portfolios are your test

[9:08 PM]Christopher Kalitin: I think the only way to really learn the way I want to is more personalised I've learned far more about engineering from my design team than the intro to engineering class

[9:09 PM]Christopher Kalitin: This blog post taught me far more about why tedious engineering processes are important than class Because in class they're only tedious, and in reality you see why they're useful first hand

[9:11 PM]Christopher Kalitin: I should write a blog post to formalize all this, it'd be a lot clearer

The reason most feedback isn't useful is that you have no way to integrate it

Ideally, feedback would give you actionable insights - the defining feature of actionable insights is that you can take action now and see the direct benefits

Without a low iteration rate, you don't actually know if what you're doing works or what insights to look for

In school you are mostly aimless aside from just taking in random knowledge. And you don't know why thing random knowledge will be applicable

If you knew directly why you needed knowledge, you'd learn far more effectively

This is clear to me on design teams where I see why understanding the hall effect is actually useful in hardware, not just learning theoretically

- - - Back to the original point, you need to be able to integrate feedback
- - - First Principles and Projects show you why the things you're learning are important!

[9:11 PM]Pieman: Yeah I agree, I assume what your getting at is this is all ineficeint [9:13 PM]Christopher Kalitin: Over the summer I found the right way to do this One new project every 2 ish weeks First, spend a week learning the theoretical fundamentals, then apply it

The faster iteration speed is extremely important and I've proven it's more useful in that I'm a better at building real programming projects than most 4th years (I talk to them on ubc solar)

- - - High iteration rate is important and motivating, in contrast to fax projects

[9:13 PM]Pieman: I would keep in mind that experiences from what you learn in school can be anecdotal bur there are some common aspects [9:14 PM]Christopher Kalitin: Yes, there's a fundamental reason for this

The more efficient humanity is, the more we flourish - you can measure this flourishing directly through GDP or net global energy output

We will conquer Alpha Centauri faster if universities are more efficient

[9:15 PM]Christopher Kalitin: I've been leaning into ancedotes, they're far more actionable than vague data that you don't actually understand

Especially on the topic of if you stay in university to learn or not, personal anecdotes are extremely important because you live a personal life [9:15 PM]Christopher Kalitin: This has actually been an immensely useful I just laid out the structure of a blog post thanks

-
    -
        - Anecdotes? Nah fuck that I'm right. I am my father's son. Symptom of extreme self belief

---

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware

Christopher Kalitin Blog

The Transition to EV Robotaxis

Apr 12, 1961 • Christopher Kalitin

1. S-curves (show them, and why they're the future) "S-curves are destiny"
2. Personal arguments do not matter, economics does
3. Implications (less car ownership, new auotmotive market decreases, majority of income from software, first mover advantage?, bankruptcies, adjacent technologies, other transport methods that can't compete, clean future)
4. Kurzweil quote about applying paradigms
5. All our problems will be solved through due to economic reasons

6. EV revolution (S-curves, declining costs, Cross over point, Graphs, companies, etc.)
7. Autonomous revolution (Predicting pace of AI, adoption in new vehicles (OEMs), retrofitting, potential first mover advantage, etc.)

The automotive market is currently undergoing a transition from privately-owned internal combustion engine (ICE) vehicles to autonomous EV robotaxis.

There are three components to this transition: ICE to EV, Manual to Autonomous, and Privately-owned vehicles to Robotaxis.

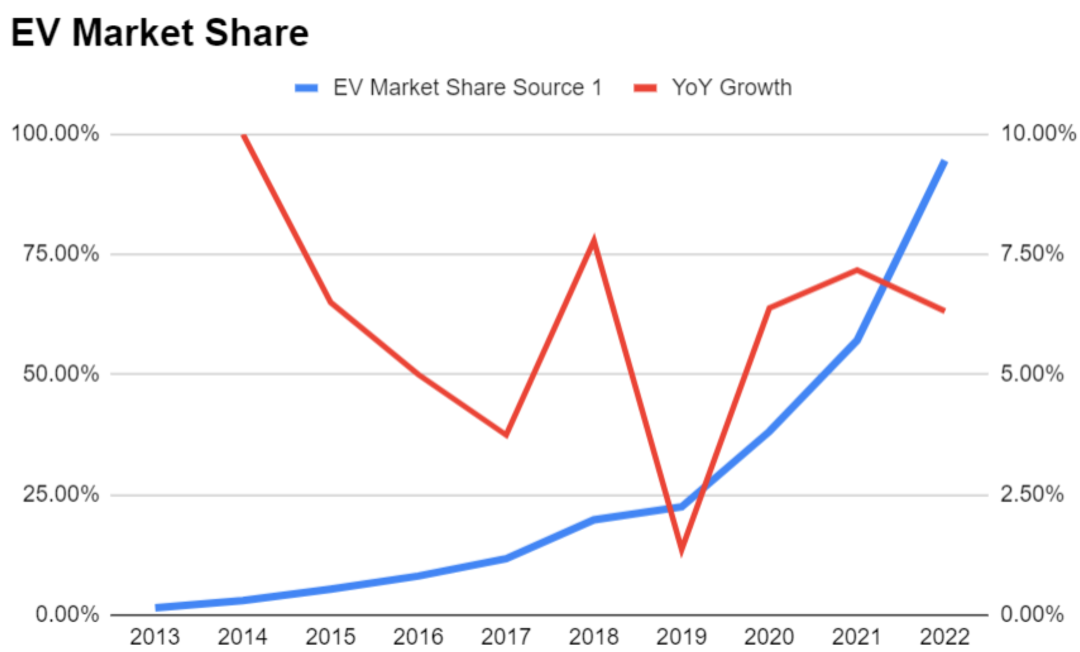
The transition to EVs is easily explained from an economic perspective.
The transition to autonomy will occur mainly due to leasure and safety reasons from the consumer's perspective.
The final transition from privately-owned vehicles to robotaxis will occur due to the improved ease of use of Robotaxis.

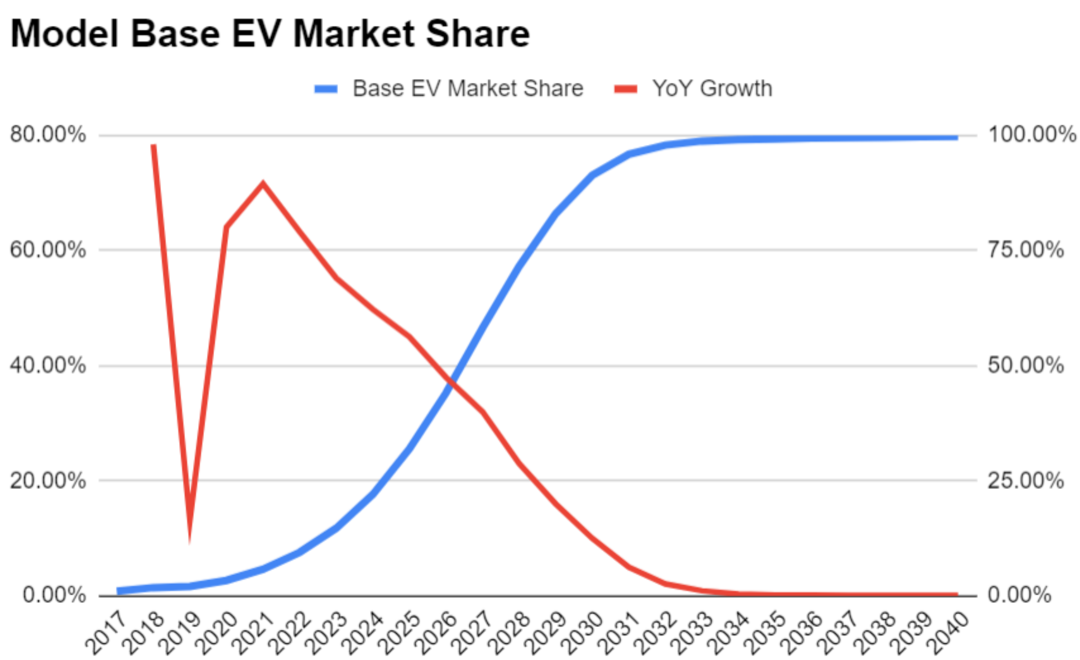
"S-curves are like gravity" - Tony Seba.
These transitions can be seen to be as likely as the sun rising tomorrow. Many society-level problems are economics problems, and economic predictions can be immensely accurate.

EV Revolution

The EV S-Curve

Currently, the EV revolution is occuring at fervent pace. This is mainly due to economic reasons and is occuring both in personally owned vehicles and commercial/heavy vehicles.





It appears growth in EV adoption has peaked in 2014 and has been sporadically declining since. This is to be expected and is visible in many other S-Curves. As any technology matures and the market becomes saturated, growth slowes. The decline in growth is not an indictment of the technology, but rather a sign of its maturity. The decline in growth serves to help identify when the technology will reach its peak market share, which in almost all cases is >99%.

Above is my prediction for EV market share until 2040, the 50% mark is in 2027 and 90% in 2030.

This is an economics problem so it can be solved with an economic toolkit. Personal arguments do not apply here, your uncle from Alabama will bregrudingly buy an EV when he realizes how much he'll save, or he'll be in the 1% of people who don't. You can not support major automakers on 1%.

Decline in EV Manufacturing Costs

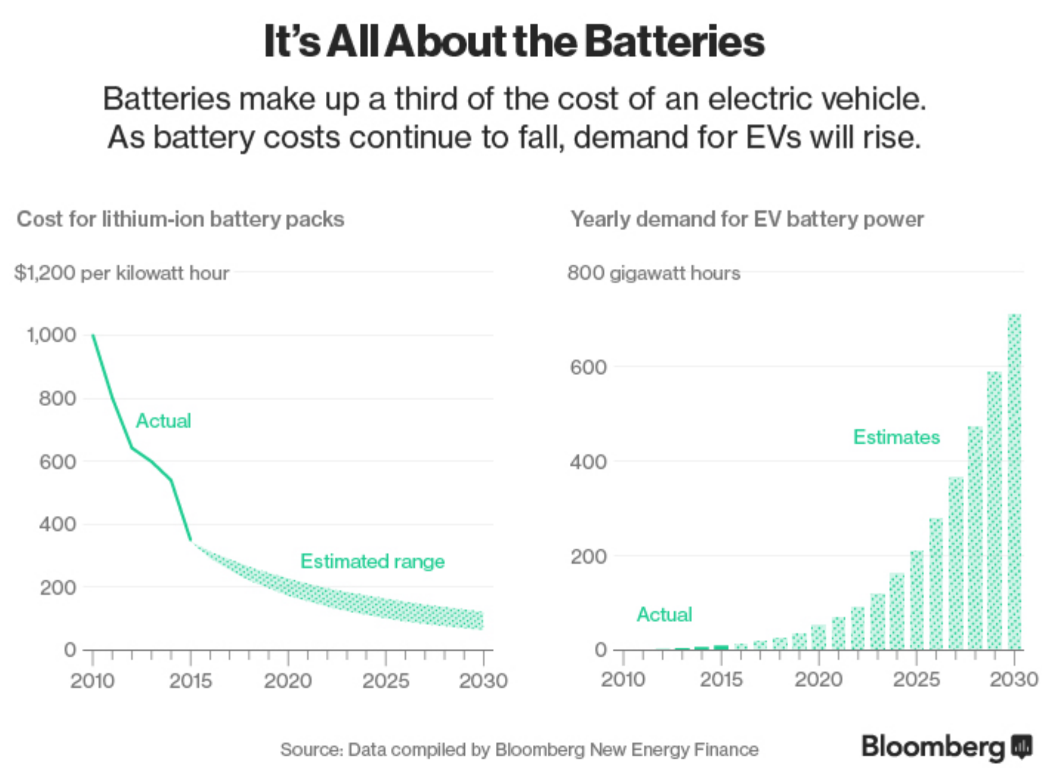
Almost all the major internal components of EVs are declining in cost.

This is most obvious in battery costs. Batteries have been on a declining cost curve for almost all time. This is due to advancements in materials science, battery chemistry, and economies of scale.

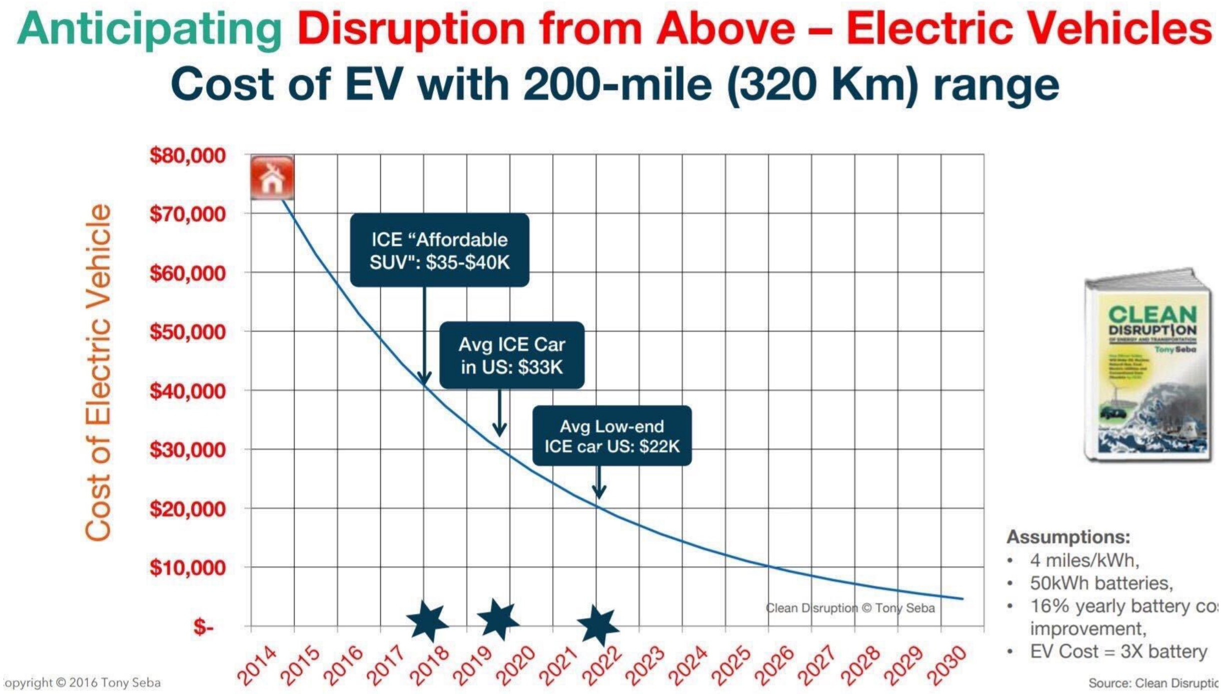
The cost of EV specific components other than batteries, such as electric motors, will also decrease in cost primarily due to economies of scale. This shows that reaching scale manufacturing of EVs is a key component to success. Speeding up transition plans from ICE to EV is the best strategy.

When taking into account all the declining costs of the underlying components, we are on track for immensely cheap EVs that out perform ICE vehicles in almost every metric. Internal combustion engines are a mature technology; we cannot hope for much more improvement.

The primary consideration when purchasing a vehicle is cost. When EVs are cheaper than all alternatives, what will happen to the alternatives?



Source of image.



Further Improvements over ICE

Decreases in purchase price are not the sole reason EVs will succeed. Operating costs, safety, and other peripheral improvements contribute as well.

It's no secret EVs have lower operating costs than ICE vehicles. Electricity costs vs. gas and less maintenance lead to overall lower operating costs. However, depreciation is a factor that can cause the total cost of ownership to increase, but this is a short term problem. As EVs gain market share due to the other factors EV-specific depreciation will cease being an issue.

The EV architecture is also inherently safer than ICE vehicles for several reasons. The battery pack moves the center of mass lower which lowers the probability of roll-over, the lack of an engine leads to a larger crumple zone, and the lack of combustion lower probability of fire.

Finally, there are peripheral improvements that effect EVs more than ICE cars. The direct to consumer model leads to a better customer experience and a streamlined purchasing process. This is far more common in the EV space due to the higher risk tolerance and ability to innovate of EV companies. The lack of a dealership also allows for a more direct relationship between the manufacturer and the consumer. This allows for better customer service and a better overall experience. The ease of charging and at home is an improvement for the majority of consumers along with the smoother and quieter ride. Furthermore, 95% of trips are under 30 miles. For these daily trips, charging at home becomes an amazing feature as you never have to spend time refueling your vehicle.

Hybrids & Hydrogen Won't Succeed Against EVs

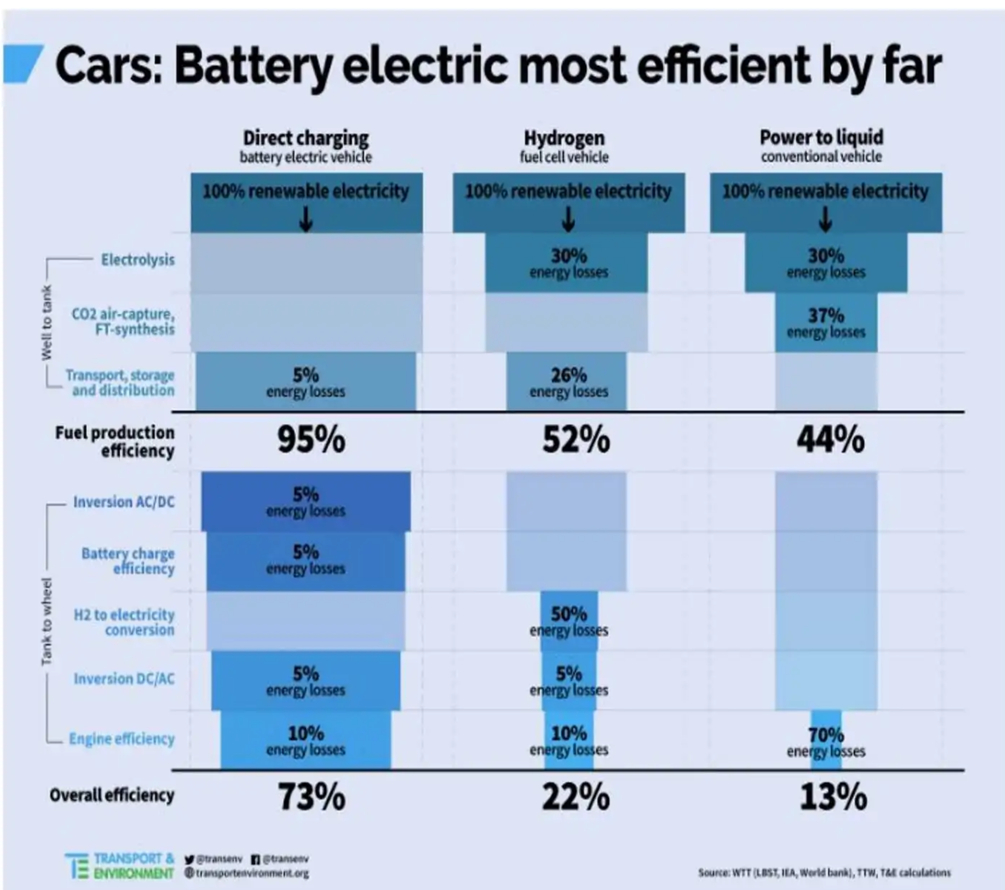
Many products that make sense from an environmental perspective do not work economically when compared to electric vehicles. This can be seen with hybrid and hydrogen vehicles.

Hybrids and Hydrogen cars do not fully benefit from the declining costs of EV technology as these designs are forced to contend with legacy and/or high-cost technology.

Again, this is mainly an economics problem and an engineering or consumer mindset may lead to incorrect conclusions.

The added range and speed of refilling of a Hydrogen vehicle is not enough to offset the cost of the Hydrogen infrastructure. The appeal of Hydrogen cars is relative to the future - mainly economic - appeal of EVs. Furthermore, with the declining costs and mass of batteries, the range and recharging speed of EVs will reach a point where the advantages of Hydrogen are negligible. To a certain extend this has already occured.

Hybrids will not be able to compete with EVs in the long term due to the static costs of the ICE powertrain, the lower maintenance cost of EVs, and decline in purchase price of EVs. Hybrids and EVs may be equal in carbon footprint, but vehicles are not purchases because of how many turtles are saved, vehcile are purchased mainly based on price.



Heavy Vehicles

The sources of decreasing cost in personally-owned EVs have vastly greater effects on heavy vehicles. Because of the increased usage of heavy vehicles and the operator's lack of consumer preferences, it is very easy to predict the success of EVs in this segment. Comparing ICE and EV semi-trucks illustrates this.

Cost of ICE Semi-Trucks	
Driver	$100k / Year
Maintenance	$10k / Year
Fuel: 200,000km / 10km/gal * $3.5/gal	$70k / Year
Other	$5k / Year
Max Payload	65,000 lb
Annual Total	$185k / Year
Annual Cost / Max Payload	**$2.85**

Cost of EV Semi-Trucks	
Driver	$100k / Year
Maintenance	$5k / Year
Fuel: 200,000km * 1.1 kWh/km * $0.16/kWh	$35k / Year
Other	$5k / Year
Max Payload	55,000 lb
Annual Total	$145k / Year
Annual Cost / Max Payload	**$2.64**

Based on the numbers I used, EV semi-trucks are currently ~7% cheaper than ICE semis when operating at maximum payload. I will use this 7% number to illustrate my points throughout this section as it is directionally correct and EVs will become cheaper over time. You can substitute your own numbers, but understanding the implications of future cost declines is the most important point.

An EV Semi saves the operator $40k (22%) per year while only decreasing payload by 10,000lb (15%). A 7% cost reduction may appear very marginal, but for large companies like Amazon and Pepsi this can save hundreds of millions of dollars per year, enough to warrant significant up-front investment in Amazon's case.

Electric semi-trucks must not exceed 82,000lb in the US. This maximum is 2,000lb more than ICE as an incentive to EV semi-manufacturers. The Tesla Semi is ~10,000lb heavier than an equivalent ICE semi and even with this drawback, operating costs are lower than that of an equivalent ICE semi. When a semi-truck is not operating at maximum load the cost improvements of EVs are even more pronounced. So, semis that often transport light loads (volume-constrained) can save more than 7% if they use EV instead of ICE.

The average cost of a semi-truck is ~$150,000. The Tesla Semi costs $150,000 for the 300-mile model and $180,000 for the 500-mile model. This is already near cost parody with ICE Semis, but EV semi-manufacturers are not done yet. Because of the decreasing costs of batteries outlined in earlier sections, EV Semis will soon be cheaper than ICE equivalents.

To illustrate the potential future cost declines in EV semis, the Tesla Semi's battery is ~900kWh. At current EV battery pack costs (not cell costs): 900kWh * $128/kWh = $115,200. Presumably, Tesla is already below this cost with the 4680 cells. If battery costs decline 80% in the next 10 years (page 8), the cost to produce an electric semi may fall by more than 2x. Even if you're less optimistic, the trend is clear.

Not only the cost to produce electric semis is decreasing, but also fuel costs. The average price of electricity in the US is $0.16/kWh. Current solar installations can produce electricity for 6-8 cents per kWh. As sustainable energy generation sources further decrease in cost and increase in production, electricity prices will fall. Furthermore, if trucks are charged mainly in the day you can forego the need for batteries. Many of industrial applications can make use of this to save costs in a future Solar, Wind, and Battery energy grid (ctrl-f "battery" to find the relevent sections).

For niche applications, electric semis are not yet ready to compete with ICE semis. In some metro areas electricity prices are absurdly high. When electricity prices are 3x the national average, electric semis are not economical. For example, if a semi's job is to offload containers from a port in a metro area, it is not ideal to use an EV semi in a poorly managed city with high electricity prices. Also, if a semi must travel >500 miles without stopping for 30 minutes to charge or if it is carrying an areodynamically inefficient payload, EV semis are not yet competitive with ICE.

The range of EV semis is not a major concern. Truck drivers drive 605-650 miles per day on average. This may require 2-3 stops to charge. If an efficient route is used, this may only delay the delivery by ~1 hour. Regardless, truck drivers legally must stop to rest and in this time the truck can be charged. Significant capital investment is required to construct the chargers, but this is a profitable business model, so it is only a matter of time before the chargers are constructed.

Semi-trucks are not the only heavy vehicles that will transition to EVs as vehicles that often travel short trips are ripe for transition to EVs. Firetrucks often only serve their local area, no more than a 10km radius. Furthermore, Buses can charge in 5-10 minutes depending on route length and battery size, short enough to charge while waiting for passengers at a home station.

Other Concerns with EVs

The largest concerns among consumers about EVs - aside from price - are range and charging time. Modern battery packs can charge from 20% to 80% in 30 minutes, and V3 Super Chargers can charge 1000 miles per hour at peak. Furthermore, 95% of trips are under 30 miles. For these short daily trips, home charging can suffice. Even a regular 110V outlet can serve the majority of use cases, though a larger home EV charger is necessary for most people ($495 from Tesla).

Because of the fast charging time of modern EV chargers, charging during road trips is not a concern. If you buy a Tesla, it will automatically route you to chargers along your trip. If you have to drive 1000 miles across the desert non-stop to escape terrorists, gas may be ideal for you. However, if you stop every few hours during roadtrips to eat or go to the bathroom in a gas car, you won't have trouble with an EV. EV faster charging companies are begining to servive the niche where they are necessary. There are interesting engineering considerations here like using an on-site battery pack to charge during peak demand.

Towing is a concern for many pickup-truck owners. In worst case condititons - cold weather, areodynamically inefficient payload - the range of your EV may be 3x lower than usual. The range of pickup trucks must be higher to accomodate this use case. The decreasing costs of batteries will allow EVs to get higher range for lower cost, but this will likely always be a weak spot for EVs. As mentioned above, it is very rare for tips to exceed 100 miles while towing. That 1% of uncles from Alabama will continue using their ICE pickups for this use case.

Modern battery packs last millions of miles. Battery health is only a concern among the absolute low-end of EVs made with bad batteries and battery lifespan is improving with each successive generation.

The grid will be able to support a fully electric vehicle fleet. If every new car sold in the US became electric tomorrow, the grid would have to grow by ~1% per year. 15,800,000 cars * 12,000 km/year * 4km/kWh = 48 billion kWh / year. The US electrical grid generated 4,231 billion kWh in 2022, growing it 48 billion kWh per year is not insurmountable. Furthermore, the lower costs of sustainable electricity generation will allow for more rapid expansion. We are entering a new paradigm of energy generation and consumption with Solar, Wind, and Batteries.

Autonomous Revolution

The Value of Autonomous Driving

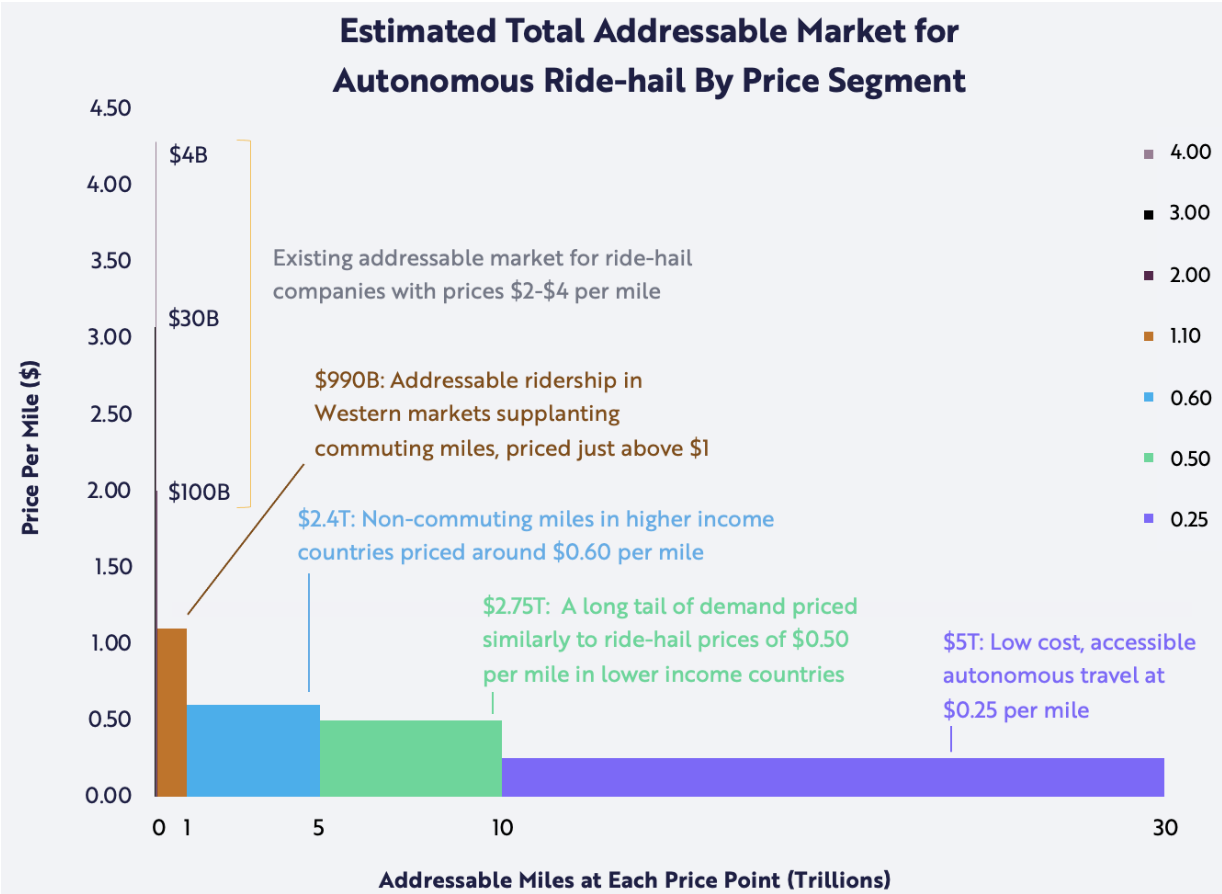
Above I detailed the two primary methods I used for analysing the value of autonomous driving. Considering the cost per hour or the cost per kilometer. The cost per hour method is more useful for selling self driving software to consumers, while the cost per kilometer method is more useful for analysing ride hailing. This section focuses on selling self driving software to consumers so we will consider cost per hour. Later, the Ridehailing section will focus on cost per kilometer.

First, cost per hour. FSD Beta currently costs $200/Month. Assuming you commute 20 hours per month (1 hour per work day), this is $10 / hour. There are two components to how enticing saving one hour of driving is to a consumer: the value of the regained time and the value of alleviating the stress of driving. A low-income teenager’s time is not very valuable and they may like driving, so the value of FSD is low. A corporate middleman’s time is more valuable and they may feel souless after commuting for a decade, making self driving car software very valuable.

Your car driving itself is a much better experience than driving manually. How much better it is depends on the time since you got your license. The most high-end experience of transportation is a luxury car with a chauffer. Self driving software allows for the chauffer element to reach every consumer.

The TAM of self driving software can be expressed on an exponential graph. A linear decrease in price leads to an exponential increase in potential customers. This can be seen in many industries ranging from the begining of the industrial revolution to the present. For example mobile internet, phones, colour TVs, radios, sewing machines, etc.

The ability to linearly decrease cost while exponentailly increasing customers allows for a higher overall profit while each subscription costs less. Creating software to drive a car is mainly an R&D project and if you are selling it directly to customers there you can take advantage of profitable device sales. The effect of this is that as total number of customers increases, R&D cost is amortized over more customers which leads to a lower cost per customer. This is the process through which the cost of self driving car software will come down in the future.



How Self Driving Will Be Achieved

Creating software that drives a car is an AI problem. It benefits from general advances in AI and the exponential progress of AI. This means you must think about self driving cars the way you think about AI, it is exponentially advancing. Would you have laughed at GPT-2 the same way people did when FSD thought the setting sun was a red light?

Everyone hoping to understand AI must read The Bitter Lesson. Over the last couple of years with Tesla switching to an end-to-end neural net approach and the failure of many self driving companies, it has become clear that feature engineering is not the proper approach to solving this problem. “The biggest lesson that can be read from 70 years of AI research is that general methods that leverage computation are ultimately the most effective, and by a large margin.” - Rich Sutton, The Bitter Lesson. Just look at the growth of AI and extrapolate into the future. But remember: it’ll never make hands, artists are safe.

The requirements of training end-to-end self driving car systems are massive amounts of useful data, massive amounts of compute, and very intelligent engineers. Tesla has the most of all of these and is most likely to succeed at creating a self-driving system first. Once other systems reach maturity and have similar capabilities to FSD they will be able to compete on cost. However, the first mover advantage will allow Tesla to partner with OEMs and install FSD in vehicles during production. Even in the case of Comma AI - a system that can be installed in any vehicle - they will be fighting an uphill battle.

Pursuing a strategy like Waymo's is very inefficient because it does not leveraging the exponential growth of compute and data. This strategy will take more resources and time to achieve the same result, by which time the competition will have already been operating at scale for years and will be well ahead. Self driving AI systems that do not leverage exponential advances in AI may succeed in the future in the same way that I could create a Chess AI that beats Magnus Carlson through sheer brute force search (depending on time control), but it would be a waste of time & money and Stockfish would easily out compete it. Assuming self driving car systems reach a plateau where they are perfect, an inefficient system will still not be able to compete due to increased costs. For example, Tesla is paid ~$20,000 for every vehicle (All of which can train and run FSD) while Waymo is paid around negative $150,000 to add a vehicle to its network.

How Much Will It Cost?

There are two components to the cost of a self driving car: the cost of hardware and amortizing development cost.

A Comma 3X costs $1250, Tesla's FSD Hardware is rumored costs a similar amount (more parts but at higher scale in a manufacturing focused organization). This cost must either be worked into the purchase price of the device/vehicle or factored into the cost of a subscription.

The development cost must also be factored in for the business to be net profitable and this will become a smaller percentage as the number of customers increases. Tesla currently spends a few billion per year on AI, at 2 million cars per year this is ~$1000 per car. Overall, it appears that the cost of development is increasing at a slower rate than the number of cars sold, so the cost per car is decreasing for Tesla. Furthermore, once true self driving is achieved, the number of customers explodes.

The hardware cost is similar to many consumer electronics devices that are sold today. Relative to a vehicle, hardware cost is negligible.

The subscription cost is much more important to consider both as a consumer and as the provider. The subscription will be the majority of the cost for the consumer. Subscriptions are 100% gross margin while hardware is far less than 100%, for this reason the subscription will be the primary source of profit for the company providing the self driving software.

Like all other subscriptions, the cost will even out when the area under the curve of customer count vs. subscription cost maxes out. I haven't found a useful enough public dataset to predict where the subscription cost will lie, but judging from other subscriptions, it could be between $50-$300/month. Tesla recently decreased the cost of FSD Beta to $100/Month.

First Mover Advantage

In the short term, the first few years after a company creates reliable self driving car software they will have a significant advantage over the competition. This first mover advantage stems from the exponential nature of AI development and difficulty of driving. Self driving is an AI problem, so it will advance exponentially. If one company has a head start, their exponential will be significant ahead of the competition. However, the first mover advantage will not last forever because the advantage of one system over another will level out as the systems approach perfect driving. For example, if one system is 99.99999% safe and the other only 99.9999%, the difference is negligible. Aside from deceptive marketing "You're 10x more likely to die in a competitor's car!", there will be no sigficant differences between the two systems.

The convergence between self driving systems is technical and the first-mover advantage stems from this. The lock-in effect will mainly be due to the difficultly of switching between different providers. This assumes technological convergence has occured. In the mean time, certain competitors will have pricing power over each other. Partnerships with OEMs will be an early indicator of a first mover advantage - of course when the software is mature enough, not before.

If devices are as easy to install and swap like a Comma 3, the effect of lock-in is diminished but remains in the cost of switching providers. You can imagine an OEM which has constructed their manufacturing lines around Tesla's FSD Hardware will be aprehensive about switching to Comma AI.

Ride Hailing Revolution

Consider this: https://x.com/aleximm/status/1867257473671082356

I started writing this blog post 293 days ago and working on it here and there over this time. I have less time than ever studying engineering at UBC, so this section will give directional analysis of where this market will go. In the future, quantative analysis is required.

This isn't a quantitative analysis, but rather thinking directionally.

THINK DIRECTIONALLY

- Cost per KM comparison to public transit ($0.1 - $0.3/km)
- Rough cost estimate: $0.1/km
- Potentially cheaper than public transit, especially for subsized public transit
- Higher efficiency than personal car and taxis
- Fleet advantage through economies of scale, but not too important

- Vehicle Market will shink (minimum number at worst rush hour)
- Not everyone will run their car as a robotaxi, market saturation. In the future, this will be a relatively low margin market per km due to competition.
- Amortize car cost

Structure:

1. Why it's a better service
2. Potential cost (Amoritzation, Insurance, maintenance, cleaning, charging, etc.) + Higher efficiency than taxis
3. Implications for public transit and current ride hailing services
4. Implications for personal car ownership and the vehicle market
5. Fleet advantage vs. personal car robotaxis

Summary

- Use economics to understand the future.

EVs:

- Declining manufacturing costs and improving technology
- Hydrogen, Hybrids, and ICE won't compete on economics
- Think directionally on trucks

Autonomy:

- Tons of value, what do you price it at?
- The Bitter Lesson, Tesla will win

Ride Hailing:

- Cheaper than personal car ownership and maybe public transit
- Will take immense market share
- The vehicle market will shrink
- Amazing service, the future looks bright

Subscribe

CKalitin

x.com/CKalitin

Geohot made a blog too. You should be working on hardware