APOLLO VIGNETTES

Allan R. Klumpp

Apollo 11 Ignores Descent Guidance Faults, Lands Anyway

During the eleven minutes of the Apollo 11 lunar landing, several alarms appeared on the display of the Lunar Module (LM). As each appeared, Buzz Aldrin, LM pilot, immediately read it aloud for Neil Armstrong, LM commander, for the team of Gene Kranz, flight director for the lunar descent in Houston, and for the designers of the descent guidance, including myself, at MIT Instrumentation Laboratory in Cambridge. None of us had any idea what caused these alarms, whether the fault was minor or a prelude to disaster. Nonetheless, Kranz directed Armstrong to press on rather than opt for safety by aborting the descent and returning to the orbiting Command Module (CM).

In ensuing days, months, and years we found out what happened. The crew's checklist called for turning on rendezvous radar during descent; it could be needed if the descent were to be aborted and the LM returned to the CM. But connections to the radar were incomplete, failing to synchronize its power supply with others. The radar's power supply drifted in and out of phase. When out of phase, the radar ate up about 15% of the guidance computer's time; there was only an 8% margin. Instrumentation Lab colleague Russ Larson now says that a time-consuming command from the astronauts aggravated the problem. As a result, the guidance computer was failing to finish its tasks, and it was complaining. The explanation became complete only this year at the design team's 25th reunion at the lab, now renamed for its founder Charles Stark Draper.

My part of the official investigation showed that throttle and steering commands, which the guidance computer was supposed to issue every two seconds, were often incompletely computed, and were queued for later completion. Any attempt to queue a command when the queue was already full (about five commands) would cause the computer to flush the queue and issue the alarm. But when the radar's power supply was in phase, queued commands, valid only at some remote past time, could be completed and issued in reverse order, momentarily taking control to guide the LM off its normal landing trajectory. Although flushing commands would cause alarms, issuing faulty commands would not. Simulations showed that faulty commands could put the LM on a crash course, and guidance would attempt to take the LM to the landing site via a trajectory that passed beneath the lunar surface.

A day or so before the reunion, the Boston Globe described an exchange between Larson and other members of the support staff at Houston as the alarms began. Not knowing what was happening, Jack Garman asked Larson what to do. Larson signaled thumbs up, Garman relayed the recommendation to Kranz, and Kranz directed Armstrong via Capsule Communicator Charlie Duke to press on.

At the reunion, I talked to Russ, and he confirmed the story. I asked what made him think the landing trajectory was safe, and he said his displays looked normal. I told him my simulations showed a crash course would look normal until it was too late. I asked why he had merely signaled thumbs up rather than giving his recommendation verbally. He said he was too scared to speak.

LM Descent Engine Throttle Incident

Although we considered an actual landing on the moon a long shot, Don Eyles and I needed some way to control the throttle of the Lunar Module descent engine in landing simulations. I proposed changing the LM throttle in proportion to the error in measured acceleration, and Don coded the throttle algorithm in the simulator. Unfortunately, delays in engine response caused the throttle to oscillate as it approached the proper setting. Engine response was modeled in the simulator as a 0.3 second delay, as specified in an Interface Control Document cosigned by NASA, by the MIT Instrumentation Laboratory where Don and I were members of the team assigned by NASA to design Apollo guidance, by Space Technology Laboratories where the engine was being built, and by the Grumman Corporation where the LM was being built.

I derived an equation to compensate for the engine delay, and Don tried it in the simulator. "The performance certainly is superior to what it was, and I had to compensate for only 0.2 seconds of the delay" beamed Don, with plots in hand. I agreed with his assessment, but wondered why he had chosen less than the optimum value for the compensation. "It's just like medicine" he said, "don't give it more compensation than it needs." I knew it wasn't just like medicine; it would be optimum to model the engine exactly. But there was a nontechnical consideration also.

Don was a new hire, fresh out of Boston University. I felt it was important to nurture self-reliance, to let coworkers' decisions on small matters prevail, even when not optimum. So I withheld my thoughts and let Don's decision stand, at least until he might reconsider it independently.

Fortunately, Don never returned to the matter. It worked well enough. Simulators at MIT/IL, at Manned Spacecraft Center, at Kennedy Space Center, and at Grumman all confirmed the performance. On 1969 July 20 Apollo 11 landed on the moon, and Apollo 12 followed in December. Both used the 0.2 second compensation for a 0.3 second delay.

Early in 1970 Grumman's Clint Tillman called. Telemetry from the Apollo 12 descent revealed erratic throttling behavior not seen in the simulators. He couldn't explain it, could I? I couldn't either, and we initiated a widespread search for the cause. A week or so later Jim Alphin from MSC called with a bombshell. Telemetry showed that the engine was four times as fast as specified. He measured a delay of 0.075 seconds, not the 0.3 second value still appearing in the Interface Control Document. Months later an outside simulation showed that a 0.075 second engine with 0.3 second throttle compensation was unstable. I verified that conclusion with our simulator and analytically as well. It became clear that had I insisted on Don coding the optimum 0.3 second compensation, Neil Armstrong and Buzz Aldrin, about a minute before touchdown, would have been propelled like a yo yo by a throttle oscillating between full thrust and idle. They probably would have aborted the first lunar landing.

(503 words)

Jim Lovell's Mission -- Apollo 13

If ever a man thumbed his nose at superstition, it was Jim Lovell, Commander of Apollo 13. And not just because of the mission number.

One of the major problems we'd had with Apollo 11 was that the computer ran out of time during the descent. This was caused by the rendezvous radar being switched on during descent, even though it was not used. The radar ate up so much computer time that there was insufficient time to complete descent guidance on many of the two-second cycles. Many guidance commands were set aside; some were never issued, others were issued out of sequence. Rendezvous radar was turned off during the Apollo 12 descent, and a more comprehensive solution was planned for Apollo 13. Even with the radar turned off, there was hardly any spare time. If the computer became overloaded, unpredictably, the Apollo 11 problems could be repeated, or worse.

For Apollo 13 and beyond, I proposed a program change, which would simply delete any guidance command that was delayed by more than a specified amount. Thus the computer to catch up, and subsequent commands would be issued more nearly on time.

After I coded it, NASA JSC first approved the change for Apollo 13, but then rescheduled introduction for Apollo 14. When I found out about this, I placed a call about 6:30 one morning to Apollo 13 Commander Lovell, who was training at Kennedy Space Center. When Dick Battin, my Boss's boss, found out about the call, he told me my political savoure faire had reached a new low. But JSC reconsidered and decided to introduce the change on Apollo 13. I was put in charge of releasing the program, which turned out to be Luminary version 131.

Apollo 13 traveled more than half way to the moon without serious incident. Then, on the Service Module, which propels the Command, Service, and Lunar modules into lunar orbit and (after lunar exploration) propels the Command and Service Modules Earthward, there was an explosion. An electric fan, enclosed for circulation in a bottle of pure liquid oxygen, developed a short circuit. In the oxygen environment, the explosion was violent.

The three astronauts and their crippled craft were left drifting helplessly toward the moon. In a heroic effort, an interorganizational team at JSC devised in a day a plan which would normally take months. The Lunar Module was made a life raft, pushing the Command and Service modules around the moon and onward to Earth. Just before the Command Module entered Earth's atmosphere, crew aboard, the Lunar Module was jettisoned. Although the changes I had made for the lunar descent were not used until Apollo 14, the rescue had been accomplished, controlled by other parts of Luminary 131. Version 131: thirteen both ways, leaving and returning.

(465 words)

The Lunar Descent Guidance Kludge*

One of the major problems with Apollo 11 was that the computer ran out of time during the descent. This was caused by the rendezvous radar being switched on even though it was not used in the descent phase. The radar ate up so much computer time that there was insufficient time to complete descent guidance on many of the two-second cycles. Many guidance commands were set aside; some were never issued, others were issued out of sequence.

The Apollo 11 crew checklist, prepared by others, not by the crew, said to switch the radar on. The checklist was wrong, but the crew did as told.

For Apollo 12, the checklist was corrected and rendezvous radar was turned off during descent. For Apollo 13, a more comprehensive solution was needed. The computer was being given new tasks without increasing the time it had to complete them. Even with the radar turned off, there was hardly any spare time. If the computer became overloaded, the Apollo 11 problems could be repeated, or worse.

For Apollo 13 and beyond, I proposed a program change, which would simply delete any guidance command that was delayed by more than a specified amount. Thus the computer could catch up, and subsequent commands would be issued more nearly on time.

After I coded the change, there was a meeting chaired by Bill Tindall at Johnson Space Center in Houston to decide about adopting my change, among others. I have always regarded Bill as the finest technical manager I have known. Bill frequently professed to know little about the technical issues involved. But it would have been humanly impossible for any one man to be expert over the wide range of issues about which Bill had to make decisions. Nonetheless, Bill invariably managed to reach a decision everyone could live with, and one with which I almost always concurred.

My proposal was approved. In the corridor after the meeting I mused to Bill, "dropping guidance commands sure is a kludge." Bill stopped in his tracks. He questioned me at length about selling his committee a kludge. Eventually, I managed to persuade him that, if we could begin anew, we would prefer either a faster computer or a longer cycle time. But both of those being impractical, my change was the best alternative.

A kludge it was, but it was introduced on Apollo 13 (which did not land on the moon) and it guided the landings of the four remaining Apollo missions without incident.

* A system composed of poorly matched components designed for some other purpose.

Tales of Pete Conrad -- Commander, Apollo 12

To document the design and operation of the Apollo primary guidance, navigation, and control system, each designer contributed a description of his/her subsystem to the GSOP (pronounced GEE-sop, for Guidance System Operations Plan). Astronauts training at the Cape for the next mission learned about the system by reading the GSOP. Likewise for astronauts of later missions at Houston. With hundreds of contributors, the GSOP grew to several hundred pages of pedantic, esoteric prose that few engineers could write and fewer could understand. For the layman, the GSOP was five pounds of Greek.

On the chance that the astronauts had not understood everything, we occasionally gave a training course at the cape. We should have boned up since we were teaching about systems we had designed a year or two before, but generally we didn't. The half a dozen or so astronauts, aware that their lives could depend upon knowing the details, were full of questions. When the designer couldn't remember an answer, Pete Conrad often could.

Four months after the first moon landing, Apollo 12 was on the launch pad. Inside the spacecraft atop the Saturn V rocket, Pete Conrad, Alan Bean, and Dick Gordon were monitoring the system. They were in reclining positions to withstand the imminent enormous thrust of the Saturn. Status lights, gauges, etc., filled their peripheral vision. Two complete guidance systems were operating. The boost rocket was controlled by its own inertial guidance located in its midsection, far from the top. The spacecraft guidance, which would convey them to the moon and back, was with the astronauts in the Command Module. Everything was running, including the spacecraft's IMU (for inertial measurement unit, a mechanical contraption comprising gyros and accelerometers that detect spacecraft motion and maintain its sense of direction). The weather was threatening, but we launched anyway.

A few seconds after launch lightning struck the spacecraft. Instantly Conrad reported that caution and warning lights lit up the spacecraft like a Christmas tree. No doubt the three astronauts expected to live only a few more seconds. But Conrad used those seconds to report the status of everything within his vision. The three had survived many emergencies staged by simsupes (simulator supervisors), but this was real.

To everyone's surprise, the Saturn V did not invert and crash. It continued upward, guided by the system in its midriff, unaffected by the lightning.

Upon reaching Earth orbit, the question was whether to continue to the moon or abort. Having transiently produced incorrect data, the spacecraft computer seemed to be working now. But the transient had tumbled the IMU, causing it to lose all sense of direction. Since the IMU was normally incapable of accurately maintaining its direction all the way to the moon and back, a sextant was aboard the spacecraft for restoring the IMU by tracking stars.

Soon Conrad, Bean, and Gordon had the ship back in order, and Conrad requested, "OK simsupe, give us the next emergency."

Lunar Module Steering

One of the great fears among the engineers who wrote the Apollo software was that an abort during a lunar descent would entail orienting the lunar module (LM) in such a way that it would lose its references and crash on the moon. This might happen even though the LM was healthy and capable of returning to the Command Module (CM). It could happen if the LM steering (a forthcoming onboard computer program that would continuously change the angular orientation of a spacecraft in response to commands from guidance) did not avoid gimbal lock, a condition I'll describe later.

At the MIT Instrumentation Laboratory (IL), now the C.S. Draper Laboratory, around 1967 we were midway in developing Apollo's guidance and navigation hardware and software. I had essentially finished the design of the Apollo Descent guidance, the computer program that would guide the LM to the landing site, and Don Eyles was translating my code to the assembly language of the Apollo Guidance Computer (AGC).

Ralph Ragan, head of the entire Apollo program at IL, summoned me and George Cherry, head of the LM software, to his office to give me an additional assignment, since the descent guidance was largely complete. The reason this assignment came from the top was that NASA had given IL an unusual directive, and Ralph wanted to make sure I complied. The assignment was to write the steering program for the LM. NASA's directive was to make no attempt to avoid gimbal lock. Several attempts had already been made for other spacecraft, and all were unsuccessful because they used too many program words and ran too slowly.

Ralph and George told me that Bernie Kriegsman (who was not present) had designed a steering program for LM, and my assignment was to code it for the AGC. Catering to my creative inclinations, Ralph permitted me to explore gimbal-lock avoidance, but only after I had Bernie's conventional design running.

Once the conventional design was running I rethought the problem. Steering a spacecraft is analogous to flying an airplane from Los Angeles to moscow. Gimbal lock is analogous to passing too close to the north (or south) pole. So, instead of taking the conventional route, a great circle, fly a route along which latitude and longitude independently progress toward the values of the destination. If neither Los Angeles nor Moscow were near a pole, the route would not pass near a pole. Such a route would never be nearer to a pole than the origin or the destination. If the destination were too close to a pole, go to the specified longitude and the maximum allowable latitude.

The conventional steering provided a standard against which the new approach could be compared. The new approach resulted in a steering system which was slightly more compact and slightly faster, and it avoided gimbal lock. Because it was better in every respect, NASA accepted it, and it landed six crews on the moon.

Near the end of Apollo, I received a phone call from an engineer at the Kennedy Space Center who did not know the LM steering avoided gimbal lock. He was trying to simulate an abort that would crash the LM, and was amazed that he couldn't make it happen.