2 Mar, 2017

Robert Trzeciak

Program Manager

Northrop Grumman IS

468 Viking Drive

Virginia Beach, VA 23452

757-498-5544 work

757-635-2628 mobile

**Progress Report – 1 Feb 2017 – 28 Feb 2017**

Contract Number: HSHQDC-06-D-00022

Contract Number 7500097279

Order Number: HSCG23-07-J-TED150

Task Order – Performance Work Statement (PWS) 1.12

Attachments: (1) SAROPS subcontractor financial reports.

1. **I noticed a few other glitches in BuildSimLand. One is that my lower resolutions aren’t really low enough. Fred’s case (see below) was taking too much time, and part of the reason was that Sim was using a low resolution, which still had too many edges. I got that fixed. I had no way of simplifying polygons with fewer than 256 edges and I got that fixed. Also, Sim was reporting an island within an island, which is an error. What happened there was that a resolution was being created with no crossings (good), but a simplifying edge was completely enclosing a small island (no crossing but bad). Although Sim handled that smoothly (it nests the polygons it uses and discards a polygon that has the same cw or ccw as its enclose), it should not have happened at all. I fixed that as well. There were some other glitches that nobody else has seen. I found them simply by carefully re-examining intermediate results (BuildSimLand generates a sequence of ShpFiles that I look at in ArcMap), and the log files that BuildSimLand and Sim produce. Along the way, I have built several LandJars; each takes a little less than 2 hours on my big machine, and a little over 3 hours on my laptop.**
2. **TRACE problem seems to have disappeared. John Squires will be running his test on each new beta to keep that problem subdued.**
3. **Worked on memory issues for Sim. John had mentioned that memory use kept growing. I noticed that, not in Sim, but in BuildSimLand. I did not want to just keep increasing the allocated memory for BuildSimLand when I knew that it shouldn’t require it. The Java garbage collector is confusing to me; Anything allocated within { and }, and not referenced outside of those brackets, should be subject to garbage collection, but for some reason, it wasn’t, and I was getting out-of-memory errors in BuildSimLand. So I went back and tried to make it “more obvious” to the garbage collector to free that memory and, to my surprise, that had an effect. I am using YourKit, a memory (and CPU) profiler), and the graphical output of that shows the effect of my efforts. So I went back and applied the same thought process to Sim. John will monitor the memory usage starting with Beta 31 (02Mar2017) to see if this has an effect.**
4. **Prepared for and attended IPR. Discussed ESS, Slippery Shore, and Planner.**
5. **Fixed bugs that were reported, amonth them, ESS, hazard+loiter. Hazard+Loiter has many possibilities and it’s hard to know if they all work. I do have 5 “control” cases that all work (an exhaustive set would have thousands), and I modified one to make a 6th. They all work. The last one yield a surprising animation, but I discussed it with two people at the IPR, and the results, after a moment or two of thought, makes sense.**
6. **Am doing a better job with “latestSeed.” I now pay attention to “latestSeed” only if I have an initial configuration for each SRU. Then I evaluate that and do NOT try to “polish” it. That saves time for Planner to work on subsequent solutions.**
7. **Fred Zummo provided a huge case to me. On the one hand, 880 time steps with 60000 particles is not reasonable, but on the other, it exposed 3 bugs. The last one was that I modified the “slippery shore” algorithm. Fred’s case wasn’t crashing (at least not within Sim; the 1 Gb particle file might have crashed something downstream), but it took hours to run. Jim noticed in the logfile that there were very many “StopAndBlock” statements. One of these is generated when Sim is blocked by a shoreline edge and tries to move the particle around it. This is not a cheap operation so I invented an algorithm that would allow a particle to slide along the shore, but if it is being repeatedly pushed into shore, the particle will simply stop for some time steps. Fred’s case works fine now (up to the creation of the particle file). To check if this is acceptable to other cases, I now have 3 test cases; one from Jim in which the currents change and the particles slide along a relatively smooth shore, one from Rob which has constant currents that (constantly) push particles into dead ends in inlets and estuaries, and Fred’s. They all work well now. In addition, the Columbia River cases still work well; the particles go downstream in an orderly manner, getting hung up only in the nooks and crannies.**
8. **Separation strategy (always PSB vs PSB is considered now) has been ironed out and seems to work well. Feb 28; Judy, Rob, Jim, and I thrashed through details on what to display and what is being computed. I’m ready to communicate more, but the end result of that meeting (phone conference) was that my part seems to be working as intended.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Activity Worked** | **Hours Worked** | **Hourly Cost** | **Total Cost** |
| Kratzke | Coding/Doc/Travel | 148 | -- | -- |
|  |  |  |  |  |
| **Totals** |  | 148 |  |  |
|  |  |  |  |  |