I think CPU heavily bottleneck after about 66k verticies, as my GPU utilization at that stage was only 27%, however my CPU was at 40%+ which is usually the uperbound CPU usage of google chrome.
A game usually renders far more than 66k verticies in a scene at 60+ fps, therefore I think the bottleneck is in the CPU. I could make this effect less drastic by writing code with less cpu intensive instructions (for loops, repeated memory access), and more gpu intensive instructions (use buffers, manipulate the particles in buffers so they can be done in batch, rather than iterating over them with the cpu outside of the buffer, then loading it back into the buffer)

Looking at the advantage that rendering triangles has, even though it requires more GPU work, suggests that the bottleneck is indeed in the CPU, as a greater number of verticies can be processed per second as a result of moving some of the work to the GPU. Overall though I am pleasantly surprised at what WebGL on a browser is able to do with my hardware configuration.

129	60		129	60
1285	60		1274	60
6425	60		10260	60
12849	60		16661	60
25700	58		23050	60
38500	47		30799	59
42405	43		38443	59
44975	37		44900	45
50115	33		51324	43
56539	26		57739	37
60395	21		66704	22
64251	17		74400	16
66821	14		80867	16
69391	14		85970	14
73245	14		89820	13
77101	13		94960	12
80955	12		100115	12
87381	11		102659	11
96375	9		106530	9