

The first test shows the time of calculation(matrix multiplication) on AWS Lambda. All matrix were initialized in functions. The operation parallelized on 100 threads due to Lambda safe limit for concurrent functions.

	1000	2000	3000	4000	5000	6000
AWS Lambda	5.76s	9.13s	21.26s	48.5s	-	-
Single node	1.49s	24.6s	114.18s	248.37s	488.62s	815.64s

At some point when you try to multiply matrix of size more then 4000, you receive exceptions. And if you try to run this test 2 times with short delay it might be the same exceptions.

The second test is the same as previous but matrix don't initialize in the body of function but they sends as parameters on server and back via HTTP response/request.

	1000	2000	3000	4000	5000	6000
AWS Lambda	80,42s	-	-	-	-	-
Single node	1.49s	24.6s	114.18s	248.37s	488.62s	815.64s

Data sends to lambda as JSON and matrix of size more than 1000 couldn't be transferred in to JSON due to lack of resources(Heap, GC).

Environment description:

- AWS Lambda:

Network speed(approximately):

Download speed: 76 Mbps

Upload speed: 94 Mbps

- Single node:

CPU: 2.7GHz Intel Core i7-4800MQ

Result values are average values of the same 10 tests in a row.