a.2

As the conclusion, it is possible to replace any state operators to another one and complete the task. For the 4 jobs in this assignment, I used these state operators:

JobSchedulingLatency: ValueState

BusyMachine: listState

MaxTaskCompletionTimeFromKafka: MapState

LongestSessionPerJob: ValueState

Part of reason why I did this is that I want to try them all. But, I think it might be a bit better to just use valueState for BusyMachine since ValueState works as a single hashmap in each window and I think the logic is easier since there is no communication between the time slots.

As for the LongestSessionPerJob, I did it a bit different. In assignment 2, I just save a hashmap where there is only maximum key value pair and output them every time the onTimer is called. Since there is always only one Value for each hashmap, ValueState is more than enough for us to use.

b.1

		FirstBack end	
	Parallel 1	Parallel 2	Parallel 4
JobSchedulingLatency	min=2, max=99, p50=50.5p99=99.0	min=79, max=169, , p50=144.p99=169.0	min=61, max=65, p50=63.0, p99=65.0
		RocksDB	
	Parallel 1	Parallel 2	Parallel 4
	min=2, max=98,p50=22.0, p99=98.0	min=26, max=11 p50=100.5 p99=111.0	min=45, max=118, p50=79.0 p99=118.0
		FileSystem	
Busy Machine	Parallel 1	Parallel 2	Parallel 4
	min=8, max=81 p50=45.0 p99=81.0	min=11, max=154p50=86.0, p99=154.0	min=17, max=151p50=49.0p99=151.0
		RocksDB	
	Parallel 1	Parallel 2	Parallel 4
	min=1, max=131 p50=60.5, p99=130.42	min=40, max=137 p50=84.0, p99=137.0,	min=112, max=142 p50=124.0, p99=142.0
		RocksDB	
	Parallel 1	Parallel 2	Parallel 4
PerMachineTaskStatistics	min=0, max=101 p50=56.0 p99=101.0,	min=43, max=194 p50=133.5 p99=194.0	min=31, max=98 p50=60.5, p99=98.0
		FileSystem	
	min=3, max=80,p50= 82.0 p99=80.0	min=14, max=191 p50=137.0 p99=191.0	min=23, max=103 p50=38.0 p99=103.00

As you can see, there is no LongestSessionPerJob in there. That is because I m encounter nullpointerexecption in the coprocession in the Collector. As the time I'm submitting this report, I haven't figure out a valid solution yet.

b.2

In general, I think there is a slight increase in lag when the parallelism increase. It is more significant from 1 to 2 than 2 to 4. Also, RocksDB generally has a better response time than in-memory state. I think there is less overhead in RocksDB than in memory. The reason might be the state size of your local disk space of the RocksDB is limited by the availability of local disk space.