TTM 591 – Supply Chain Management

Sourcing Simulator Project

Littlefield Technologies Game 2

# Project guide- Dr. Kristin A. Thoney

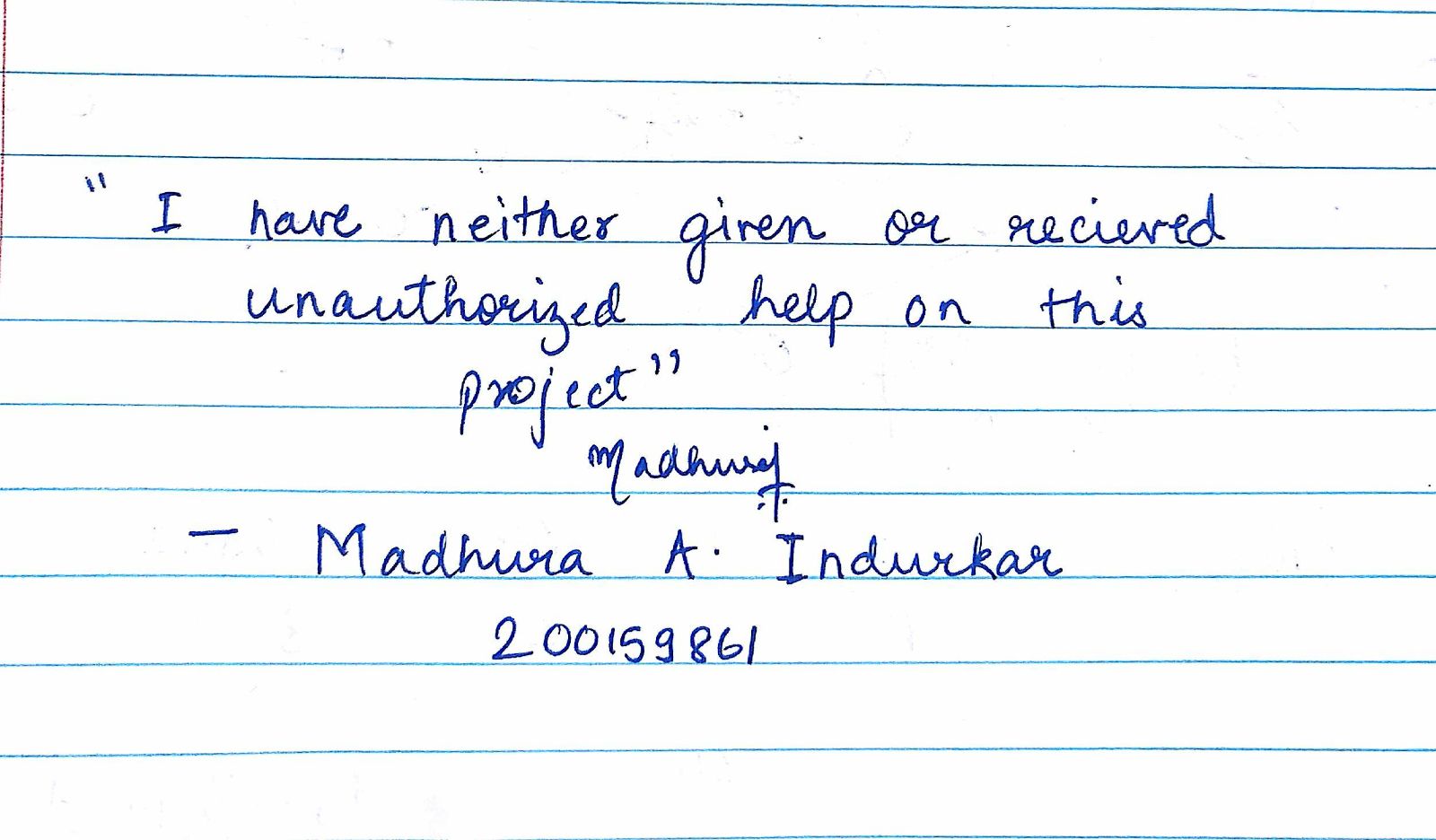
Project Report by-

**Siddharth Rastogi**

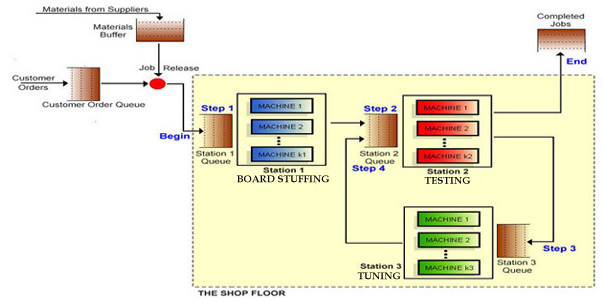
001004393

**Madhura Indurkar**

200159861

**INTRODUCTION**

At the beginning of the game, all of the jobs on all 3 stations were sequenced according to “First in First out” (FIFO) policy which was the default setting of the game. As shown in **Figure 1**, Station 1 is the Board Stuffing Station, Station 2 is the Testing Station and Station 3 is the Tuning Station.



**Figure 1: The Shop Floor**

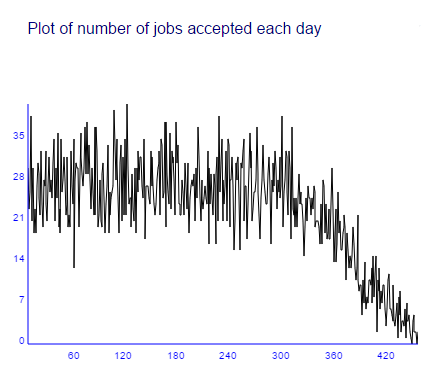
The incoming jobs were sequenced according to the following steps (**Figure 1**):

Step 1: Mounting and soldering electronic components onto PC boards at the board stuffing station (Station 1).

Step 2: The second step collects test data on each Receiver at Station 2 that is transmitted to tuning for the third step.

Step 3: The third step tunes the DSS units to receive satellite signals at the tuning station (Station 3). Step 4: The final step returns boards to the testing station (Station 2) t]for a final burn-in and certification required by customers.

At the beginning of the game, the numbers of machines provided initially were insufficient to complete jobs to cater the demand. Moreover, the cash available initially was insufficient. Our initial strategy at the beginning of the game was based on the initial average demand for 10 days (which was 25.4 jobs per day) and the processing times of the machines obtained from calculations done after Littlefield Game 1. Therefore, we calculated interarrival time based on 25.4 jobs per day. Accordingly, interarrival time is 0.9448 hrs or 57 mins (24/25.4 x 60). Moreover, average demand for first 286 days was expected to be constant. As we can see in the demand graph (**Figure 2**), the demand (number of jobs accepted per day) was moreover constant through first 286 days of the game. Therefore, we considered simulation end time as 6864 hrs (286 x 24). Based on simulation 1 HW, processing time for each machine in station 1 is 3.46 hrs, in station 2 is 1.165hrs and in station 3 is 2.74 hrs.

****

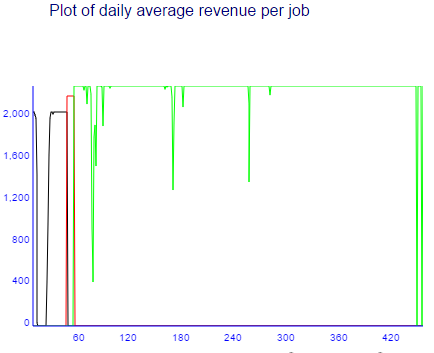
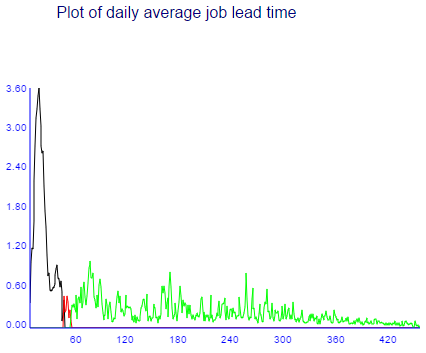
**Figure 2: Job arrivals (Demand)**

Based on these numbers, we we made several decisions throught the game to increase our total cash in hand. Details on our decisions and logic behind them have been discussed in details in the following section.

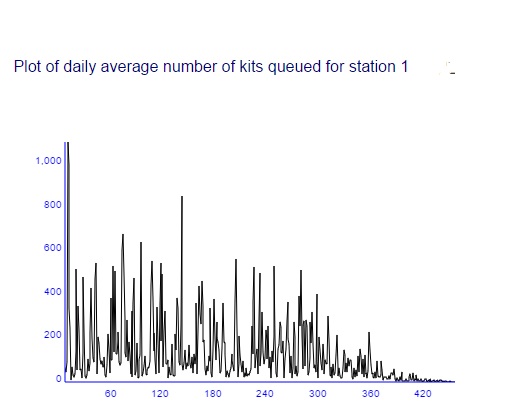
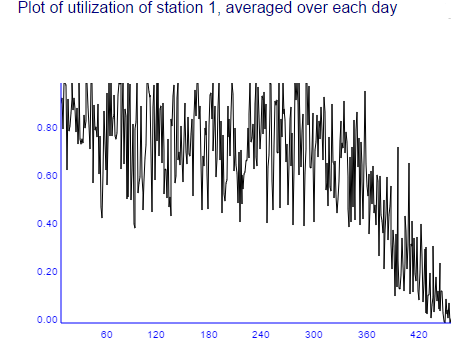
**REVIEW OF DECISIONS**

In combination with SAS simulation, we used graphs on utilization of each station (**Figure 4, Figure 5, Figure 6**), daily average number of kits queued for each station (**Figure 4, Figure 5, Figure 6**), daily average revenue per job (**Figure 3**) and daily average job lead time (**Figure 3**) to identify the bottlenecks of the factory and also to make rough estimates of the number of machines we would require at each station. On observing the utilization plots of the workstations, we identified that workstations 2 and 3 as bottlenecks of the system and decided that we would need to buy more of machines 2 and 3 after the game starts. In order to have a fairly better idea of the number of machines that should be bought after the start of the game, we ran 3 simulations of our model on SAS Simulation Studio. We simulated our model using 10 replicates and the process parameters from Game 1 and obtained the results for the following parameters:

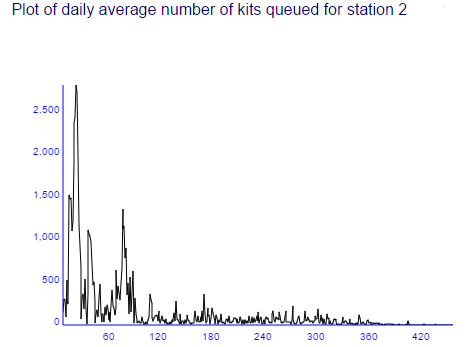
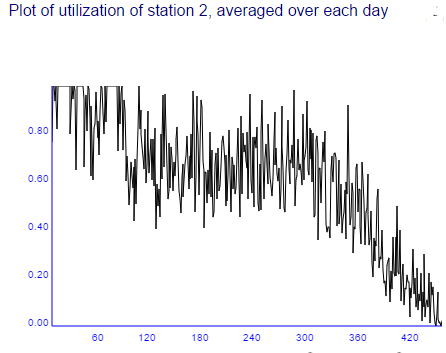
1. Average Machine Utilizations
2. Average and maximum waiting time for a station
3. Average time in the system

****

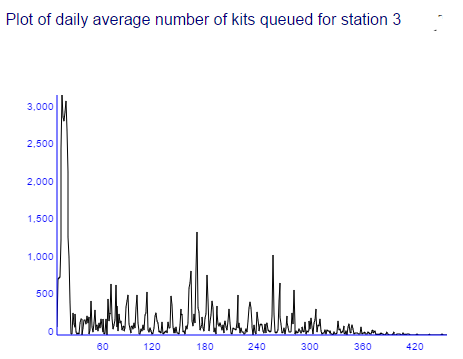
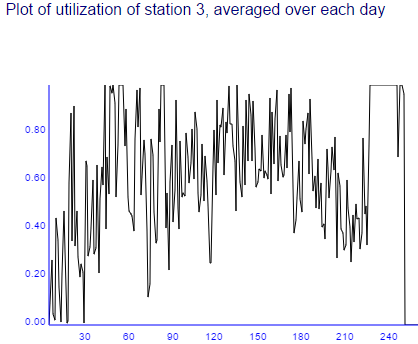
**Figure 3: Daily average job lead time and daily average revenue per job**



**Figure 4: Utilization and daily average number of kits queued for station 1**

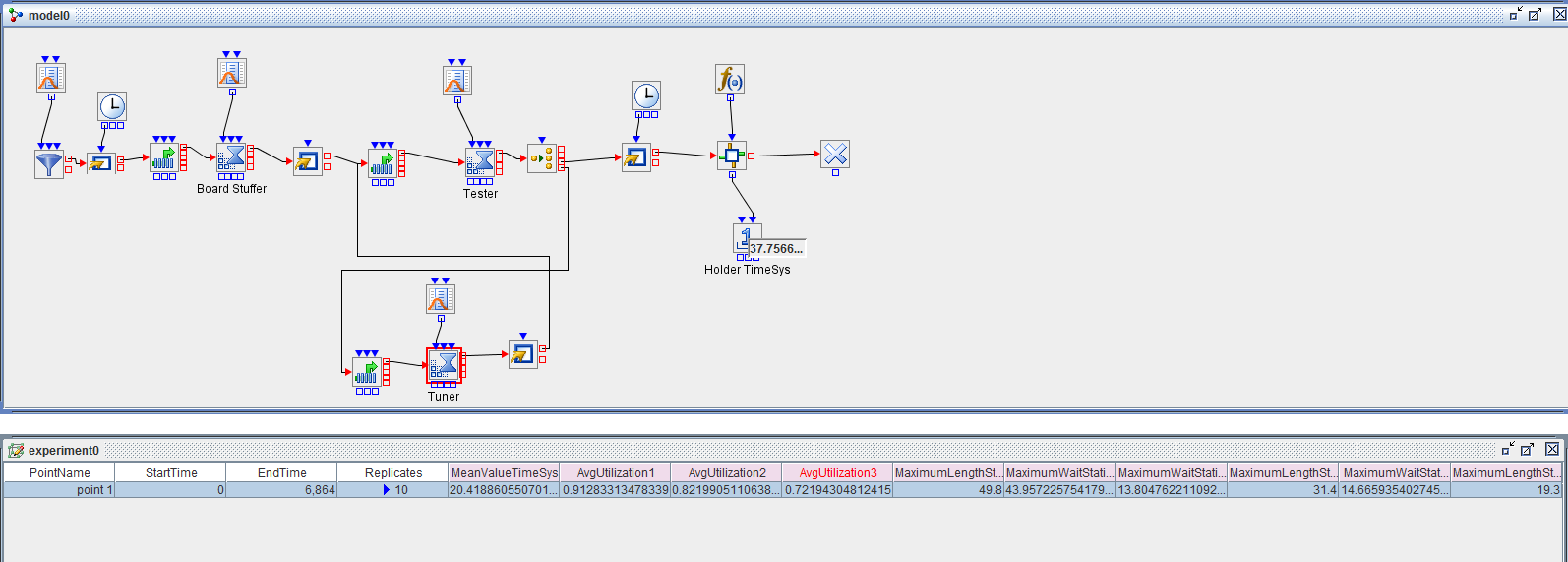


**Figure 5: Utilization and daily average number of kits queued for station 2**

****

**Figure 6: Utilization and daily average number of kits queued for station 3**

As discussed in our initial plan, we ran three simulations to decide how many machines for station 2 and station 3 should we buy. Our final simulation (**Figure 7**) suggested to buy one machine for station 2 and two machines for station 3. Specifically, running third simulation with 3 machines in station 2 and 4 machines in station 3, we find following results: Average utilization for Station 1 in long run is expected to be 0.91 but for station 2 to be 0.82 and station 3 to be about 0.72. Moreover, maximum weight time for station 1 is 43.9 hrs (1.82 days) for station 2 is 13.80 hrs (0.57 days) and for station 3 is 14.66 hrs (0.61 days) respectively. In addition, **average time in the system is 20.41 hrs (0.85 days).** In conclusion, meeting delivery time of 1.5 days was possible in this scenario.



**Figure 7: SAS Simulation results of our final simulation**

Specifically, we continuously monitored these parameters to make our decision of buying and selling machines. Our strategy was as follows: At the start of the game, we decided to buy 1 m/c for station 2 ($150,000 each) and 2 m/c for station 3 ($140,000 each) based on our Simulation results and utilization graphs of the stations. For which we needed to borrow $430,000. Since we had less than $200,000 Cash to pay for next order and our calculated ROP was 140 jobs (8600 kits) we decided to take a loan of $630,000 to account for the machines and material order costs. For that we needed to pay instant $31,500 loan processing fee from the cash we had then. We made buying decisions for stations 2 and 3 immediately after and also changed our reorder point and increased order quantity accordingly as per our calculations to satisfy the demand.

As the game progressed, we started monitoring utilization and average number of kits queued for all stations (**Figure 4, 5 and 6**). We made buying decisions for station 1 (**Figure 4**) when its utilization started approaching 1 and average number of kits queued for station 1 started increasing (**Figure 4**). But in order to do that, we had to borrow more money due to shortage of cash, but we decided to do so as lost profits due to increased lead times or lost orders would have accounted for more than the amount spent in loan interest and processing fees of the loan.

Specifically, even after buying one machine for station 2 and two machines for station 3 on day 10, the lead times of the completed jobs continued increasing (**Figure 3**) as the game progressed during initial stage of the game and the jobs continued to queue in for machines of station 1, 2 and 3 (**Figure 4, Figure 5, Figure 6**). As the lead times of completed jobs increased, there was a dip in the overall revenue (**Figure 2**). Due to this, we decided to buy one machine for station 1 on day 22 because station 1 was the bottleneck as identified by SAS simulation (**Figure 7**).

Moreover, at the start of the game, all of the jobs on station 2 were sequenced according to “First in First out” policy which was the default setting of the game. Specifically, there was no priority given either to step 2 or to step 4 at station 2. However, we decided to give priority to Step 4 instead at day 50 (**Table 1**). By doing so, the jobs that were waiting to be sent to the customers after the final burn-in and certification started receiving priority, thus, reducing the lead time for the customers. However, we agree that in long term both strategies will give almost same return. But, in the initial stage of the game, the state of the factory was unstable due to excessive pending orders. Therefore, our strategy to give priority to Step 4 at day 50 (**Table 1**) was affected by the state of the factory at the beginning.

Overall, our decision for buying one more machine for station 1 was impacted by the state of the factory at the beginning of the game. Moreover, we changed our order quantity to 9840 kits on day 46. Our order quantity calculation was also impacted by the initial lack of cash because of which we had to take loan whose interest rate was 15%. So, our total cost of capital became 23% (15 +8). On day 46, we used Economic Order Quantity (EOQ) formulae to calculate our order quantity. Following values were used in the EOQ formulae:

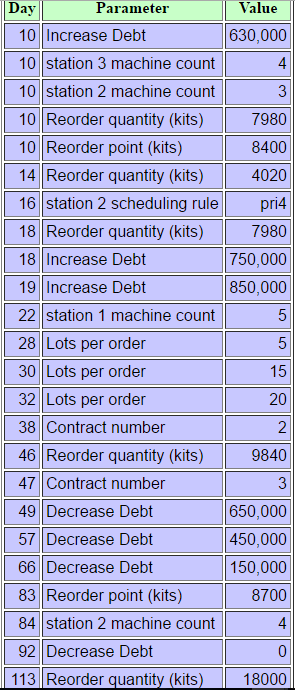
D =Total Demand in 365 days (in kits) = 25.4 jobs per day x 60 kits per job \* 365 days = 556, 260 kits

C = Ordering cost per order = $500 per order

Ic = Inventory holding cost per kit per year = 23% of $25 = 5.75

EOQ = Square Root (2DC/Ic) = 9836 kits = 164 jobs

We also increased the lot sizes gradually from 5 (initial) to 20 to experiment which lot size will be more efficient. Through our experimentation, we find that lots per order of 20 will help to provide the least possible job lead times. Therefore, we kept our lots per order of 20 throughout the game after changing it to 20 on day 32. As can be seen from our transaction history (**Table 1)**, Based on that we also changed our contract to 2 (day 38) and later to 3 (day 47) because our job lead times were sufficiently low to meet needs of our customer within 0.75 days.



**Table 1: Transaction History**

However, even after buying one machine for station 2, one machine for station 1, and two machines for station 3, the lead times of the completed jobs continued increasing between day 75 and day 85 (**Figure 3**) and the jobs continued to queue in for machines of station 2 (**Figure 5**) because station 2 was the bottleneck as identified by SAS simulation. As the lead times of completed jobs increased, there was a dip in the overall revenue (**Figure 2**). Due to this, we decided to buy one machine for station 2 on day 84 (**Table 1**) because station 2 was the bottleneck.

A total of 5 machines of station 1, 4 each of stations 2 and 3 were sufficient to cater for the demand till it started decreasing, without the machines being over utilized, the revenue remaining constant and lead times fluctuating but not much. As the game progressed, we did not need to buy more machines after executing our initial buying strategy as the average demand was more or less constant. As our revenue increased, we started to pay off the loan that we had taken.

On day 113, when our revenue stream stabilized and we paid off all our debt, we used Economic Order Quantity (EOQ) formulae again to calculate our order quantity. As we paid off all our debt, we used cost of capital of 8%. Following values were used in the EOQ formulae:

D =Total Demand in 365 days (in kits) = 25.4 jobs per day x 60 kits per job \* 365 days = 556,260 kits

C = Ordering cost per order = $500 per order

I = Inventory holding cost per kit per year = 8% of $25 = $2

EOQ = Square Root (2DC/Ic) = 16, 677 = 278 jobs

However, since we were already in 113 day and only 173 days of game was left when average demand was expected to be 25.4, we decided to keep order quantity of 300 jobs or 18,000 kits throughout rest of our game.

Our Order Quantity Strategy, in which we choose our EOQ of 18000 kits from day 113, worked well. Specifically, we spend less in total inventory cost than would have spent if continued with initial order quantity of 8000 kits. Sticking to initial order quantity of 8000 kits might have costed more in ordering costs than the savings that could be made from holding less inventory. Following calculations illustrates our point:

Order quantity kept from day 113 to throughout rest of our game was 18000 and we got total of 24 shipments during this period. In total, we ordered 432,000 kits (18000 x 24 orders) in this period with the following ordering costs and inventory holding costs (As we paid off all our debt by day 113, we used cost of capital of 8%):

Total ordering cost = $500 x 24 orders = $12000

Total holding cost = (18000/2) x (8% x $25) = $18000

Total product cost = $25 x 432,000 = 10,800,000

Therefore, total inventory cost is equal to $10,830,000. However, if we would have continued with the initial order quantity of 8000 kits, it would have resulted into 54 orders (432,000/8000). Therefore, following costs will be incurred if we would have continued with the initial order quantity of 8000 kits from day 113 to throughout rest of the game (As we paid off all our debt by day 113, we used cost of capital of 8%):

Total ordering cost = $500 x 54 orders = $27000

Total holding cost = (8000/2) x (8% x $25) = $8000

Total product cost = $25 x 432,000 = 10,800,000

Therefore, total inventory cost that we would have incurred if we would have continued with the initial order quantity of 8000 kits from day 113 to throughout rest of the game is equal to $10,835,000 Since total inventory cost of $10,835,000is higher that total inventory cost of $10,830,000, we believe that our EOQ of 18000 kits from day 113 to throughout rest of the game is much better that initial order quantity of 8000 kits. Therefore, we believe that our Order Quantity Strategy, in which we choose our EOQ of 18000 kits, worked well.

During the latter part of the game, as the demand started decreasing, we decided not to sell any machine due to very low returns from selling machines. After Game 1, the loss of revenue due to missed orders in the end part of the game, we had concluded that the selling out of machines to generate revenue from them was an incorrect decision and we could have generated greater revenue had we not missed out on the orders due to insufficient number of machines. Therefore this time, we decided not to sell out the other machines (learning from the mistake we made in Littlefield Game 1) as we could not correctly estimate the exact demand and wanted to avoid losses due to lost orders or increased lead times till the game ended. In conclusion, our strategy of buying and selling machines worked pretty well for us in Game 2 and SAS Simulation Studio proved to be an extremely helpful tool in deciding the count of machines to be bought based on the station utilizations and queues.

As part of our initial game strategy, we calculated our ROP including some safety stock to be 140 jobs (8400 kits), given the average daily demand to be 25.4. Specifically, to calculate our Reorder Point (ROP), we used the following formula:

**ROP = Average Demand x Lead time + Safety Stock**

Directly using the above formula was not found feasible given high variation in demand. Therefore, we used following formula and used safety stock of 10 jobs per day making maximum demand of 35 jobs per day (25 average daily demand +10 units of safety stock per day).

**ROP = Maximum Demand x Lead Time**

Our ROP strategy worked well. Specifically, we wanted to provide high customer service in the range of 95% to 98%. By using Maximum demand in our ROP formulae in place of Average demand that is used when demand is constant, we tried to cover demand uncertainty by maintaining some extra safety stock. Specifically, we expected that demand during lead time of four days should be lower than 4 x Maximum demand. In fact, as per our expectation, we never had negative inventory throughout the game. Overall, with our ROP strategy, there was very little chance to have lower inventory of parts than the outstanding orders from customers.

**CONCLUSION**

At the beginning of the game, the numbers of machines provided initially were insufficient to complete jobs to cater the demand. Moreover, the cash available initially was insufficient. Our initial strategy at the beginning of the game was based on the initial average demand for 10 days (which was 25.4 jobs per day) and the processing times of the machines obtained from calculations done after Littlefield Game 1. Based on these numbers, we we made several decisions throught the game to increase our total cash in hand. Details on our decisions and logic behind them have been discussed in this paper.

A total of 5 machines of station 1, 4 each of stations 2 and 3 were bought to meet demand of our customers and were found sufficient to cater for the demand till it started decreasing, without the machines being over utilized, the revenue remaining constant and lead times fluctuating but not much. As the game progressed, we did not need to buy more machines after executing our initial buying strategy as the average demand was more or less constant. As our revenue increased, we started to pay off the loan that we had taken. This Strategy of buying and selling machines worked pretty well for us in Game 2 and SAS Simulation Studio proved to be an extremely helpful tool in deciding the count of machines to be bought based on the station utilizations and queues.

Our Order Quantity Strategy, in which we choose our EOQ of 18000 kits from day 113, worked well. Specifically, we spend less in total inventory cost than would have spent if continued with initial order quantity of 8000 kits. Sticking to initial order quantity of 8000 kits might have costed more in ordering costs than the savings that could be made from holding less inventory. Specifically, total inventory cost that we would have incurred if we would have continued with the initial order quantity of 8000 kits from day 113 to throughout rest of the game is equal to $10,835,000. Since total inventory cost of $10,835,000 is higher that total inventory cost of $10,830,000 that we incurred with EOQ of 18000kits, we believe that our EOQ of 18000 kits from day 113 to throughout rest of the game is much better that initial order quantity of 8000 kits. Therefore, we believe that our Order Quantity Strategy, in which we choose our EOQ of 18000 kits, worked well.

As discussed above, our ROP strategy also worked well. Specifically, we wanted to provide high customer service in the range of 95% to 98%. By using Maximum demand of 35 jobs per day in our ROP formulae in place of Average demand of 25.4 jobs per day that is used when demand is constant, we tried to cover demand uncertainty by maintaining some extra safety stock. Specifically, we expected that demand during lead time of four days should be lower than 4 x Maximum demand. In fact, as per our expectation, we never had negative inventory throughout the game. Overall, with our ROP strategy, there was very little chance to have lower inventory of parts than the outstanding orders from customers.

Finally, our choice of contracts was also good. We choose contract 3 as our final contract because it earned the most money but required lowest job lead time among all contracts of 0.75 days. To reach to our final decision, we increased the lot sizes gradually from 5 (initial) to 20 to experiment which lot size will be more efficient. Through our experimentation, we find that lots per order of 20 will help to provide the least possible job lead times. Therefore, we kept our lots per order of 20 throughout the game after changing it to 20 on day 32. As can be seen from our transaction history (**Table 1)**, Based on that we also changed our contract to 2 (day 38) and later to 3 (day 47) because our job lead times were sufficiently low to meet needs of our customer within 0.75 days.