College of Engineering

UMASS COLLEGE OF ENGINEERING

INDUSTRIAL ENGINERRING

Supply Chain Logistics M&I-ENG 578 (153206) SP24

NAME OF THE PROJECT: Assembly Systems Project

SEMESTER: Spring 2024

NAMES: Il Chang, Nicolas Boada, Rutuja Sachin Ajagekar, Sarthak Raj

PROFFESOR: ANA MURIEL

DATE: 04/11/2024

Introduction

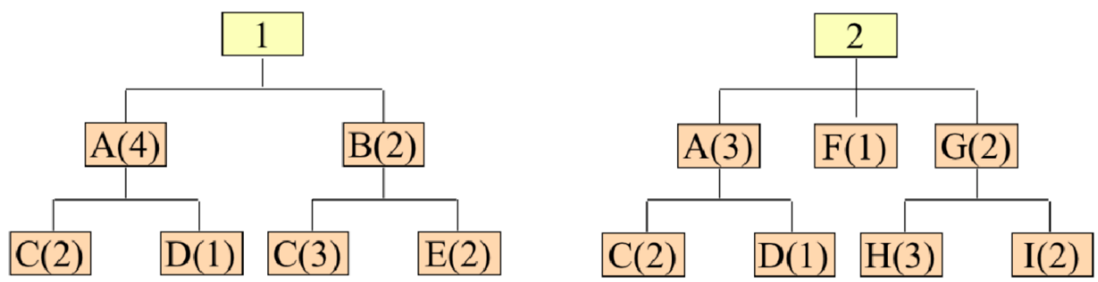
This project worked on short-term recommendations to improve the scheduling of an assembly line for two products, each with different parts. The improvements need to happen quickly because Jet Engine, the company behind this project, makes both commercial and military engines. These engines not only need a lot of work to produce but also require complex logistics to assemble efficiently. This aims to cut costs and meet demand better.

The project evaluated various aspects, starting from a simple assembly line model to a more complex one. It involved considering setup costs, specific demands, and finding different solutions based on the variables in the model. Finally, it offered suggestions for better long-term performance by testing various capacity values in the most challenging model.

Question 1

Write an AMPL data file that captures the given information on the BOM and on demand.

To check the build of material the image below presents the process of building the two engines.



Graphic 1. Product tree diagram for the two leadings products

The first thing we notice is that we have a set of parts. In the data file, this set is named set I and includes 1, 2, A, B, C, D, E, F, G, H, I. These represent engine 1 and engine 2, along with the parts needed to build them. Then, the engines need to be produced over a certain period of time. In this case, a parameter "n" was defined to specify the number of weeks for planning the demand.

The next step is defining the build of material, for a better understanding the BOM is presented in the table below.

Part needed	Build part	Quantity needed
A	Engine 1	4
B	Engine 1	2
C	A	2
C	B	3
D	A	1
E	B	2
A	Engine 2	3
F	Engine 2	1
G	Engine 2	2
H	G	3
I	G	2

Table 1.BOM Table

The only part left is setting the demand, for that a parameter "D" was created. The demand considered is the one shown in the figure below.

	Week									
End Item	1	2	3	4	5	6	7	8	9	10
1	5	5	5	8	8	8	8	8	5	5
2	5	5	5	8	8	8	8	8	5	5

Figure 1. Specify Demand for Engine 1 and 2

Additionally, there is a specific requirement of 20 units of part C on week 3, 8 units of part D on week 5 and 2 units of part F on week 7.

The complete data file is shown below.

```
set I:= 1, 2, A, B, C, D, E, F, G, H, I;
param n:= 10;

param BOM:=
A 1 4
B 1 2
C A 2
C B 3
D A 1
E B 2
A 2 3
F 2 1
G 2 2
H G 3
I G 2;

param D: 1 2 3 4 5 6 7 8 9 10:=
1   5   5   5   8   8   8   8   8   5   5
2   5   5   5   8   8   8   8   8   5   5
A   0   0   0   0   0   0   0   0   0   0
B   0   0   0   0   0   0   0   0   0   0
C   0   0   20  0   0   0   0   0   0   0
D   0   0   0   0   8   0   0   0   0   0
E   0   0   0   0   0   0   0   0   0   0
F   0   0   0   0   0   0   2   0   0   0
G   0   0   0   0   0   0   0   0   0   0
I   0   0   0   0   0   0   0   0   0   0
H   0   0   0   0   0   0   0   0   0   0
;
```

Figure 2. Q1 and 2 Assembly Data

Question 2

Write and run an AMPL model to calculate all of the parts that will need to be produced and when to satisfy those demands assuming zero lead times and no capacity constraints.

Considering the data file from question 1, sets I, parameter n, parameter BOM, and parameter D are set up in the same way. Additionally, a parameter T is added ranging from 1 to n, representing the planning horizon. BOM specifies the quantity of engines and parts required for assembly. Parameter D represents the demand for component i at time t.

Then, an integer variable is created to monitor the production process of component i at time t. This variable is integer because it deals with exact quantities like whole engines or parts, not fractions. We're not considering Work In Progress (WIP) because we want precise production quantities. The objective function aims to minimize production while meeting demand as much as possible. The model is shown in the following figure.

```
reset;

set I; #Set of number of parts
param n; #Number of weeks of planing demand
set T= 1..n;# Planning horizon
param BOM{I, I} default 0; /* Bill of materials child parent quantities*/
param D{I,T} default 0; /* Demand for component i at time t */

var X{I,T}>=0 integer; /*Production of component i at time t*/

minimize Production: sum{i in I, t in T} X[i,t];

subject to MeetingDemand{i in I, t in T}:  X[i,t]=D[i,t]+sum{j in I} BOM[i,j] * X[j, t];

data Assembly1.dat;
option solver cplex;
solve;
display X;
```

Figure 3. Q 1 and 2 Assembly Model

After running the model, the results given are the following.

```
Solution determined by presolve;
objective Production = 3605.
X [*,*]
:    1     2     3     4     5     6     7     8     9    10    :=
1     5     5     5     8     8     8     8     8     5     5
2     5     5     5     8     8     8     8     8     5     5
A    35    35    35    56    56    56    56    56    35    35
B    10    10    10    16    16    16    16    16    10    10
C   100   100   120   160   160   160   160   160   100   100
D    35    35    35    56    64    56    56    56    35    35
E    20    20    20    32    32    32    32    32    20    20
F     5     5     5     8     8     8    10     8     5     5
G    10    10    10    16    16    16    16    16    10    10
H    30    30    30    48    48    48    48    48    30    30
I    20    20    20    32    32    32    32    32    20    20
;
```

Figure 4. Q 1 and 2 Results

Question 3

Assume now a lead time of one week for each step. Change your model to consider this. Will your data file need to change as well?

In this case, the model remains practically the same; the identical dataset is utilized, with the addition of a parameter, L, representing lead time, and a parameter, A, quantifying the number of elements previously existing in the system. To incorporate these additions into the constraints, an additional lead time constraint is introduced, and the parameter L is utilized to bound the planning horizon parameter T, thereby accounting for lead time. The model is presented below.

```
reset;

param n;
set I;

set T=1..n;
param BOM{I,I} default 0;
param D{I,T} default 0;
param L{I} default 1;
param A{I,T} default 0; #available quantity of part i at time t


var X{I,T} >=0 integer;

minimize production: sum{i in I, t in T} X[i,t];
subject to BoM {i in I, t in T: t>L[i]}: X[i,t-L[i]]= D[i,t]+sum{j in I} BOM[i,j]*X[j,t];
subject to LT {i in I, t in T: t<=L[i]}:  A[i,t] = D[i,t]+sum{j in I} BOM[i,j]*X[j,t] ;

data Assembly1.dat;
option solver cplex;
solve;
display X;
```

Figure 5. Q 3 Assembly Model with Lead Time

After running, the model the answer is presented below.

```
presolve, constraint LT[1,1]:
        no variables, but lower bound = 5, upper = 5
presolve, constraint LT[2,1]:
        no variables, but lower bound = 5, upper = 5
presolve, constraint LT[2,1]:
        all variables eliminated, but lower bound = 5 > 0
presolve, constraint LT[1,1]:
        all variables eliminated, but lower bound = 5 > 0
presolve, constraint BoM[2,3]:
        all variables eliminated, but lower bound = 5 > 0
3 presolve messages suppressed.
X [*,*]
:    1     2     3     4     5     6     7    8    9   10    :=
1    0     0     8     8     8     8     8    5    5    0
2    0     0     8     8     8     8     8    5    5    0
A    0    56    56    56    56    56    35   35    0    0
B    0    16    16    16    16    16    10   10    0    0
C  160   180   160   160   160   100   100    0    0    0
D   56    56    56    64    56    35    35    0    0    0
E   32    32    32    32    32    20    20    0    0    0
F    0     8     8     8     8    10     5    5    0    0
G    0    16    16    16    16    16    10   10    0    0
H   48    48    48    48    48    30    30    0    0    0
I   32    32    32    32    32    20    20    0    0    0
;

production = 2780
```

Figure 6. Q3 Results

There is a difference between production when considering lead time and when not. In the first model, the production value is 3605 units, whereas in the second model, it is 2780 units. This indicates that incorporating additional lead time results in 22.88% less production. However, adding more constraints makes it a more realistic and robust model.

Question 4

Change your model to incorporate the processing times and capacity constraints. Find the lowest cost production schedule given the holding costs (H) and penalties (P) for each part.

To solve this problem, the model requires changes, which are outlined below:

- Parameters added:
  - Parameter P: Represents the penalty for part I.
  - Parameter H: Denotes the holding cost of part I.
  - Parameter R: Signifies the processing time of part I in the bottleneck.
  - Parameter C: Represents the total capacity in terms of the time period.

- Sets added:
  - Set K: Represents the number of parts that pass through the bottleneck.
  - Set T0: Represents the consideration of a 0 week with some material available.

- Variables added:
  - Variable F: Represents the fulfilled customer demand of part i at time t.
  - Variable U: Represents the cumulative unfulfilled customer demand of part i at time t.
  - Variable V: Represents the inventory of part I at time T.

In addition to these elements, the constraints were modified and configured with new parameters to ensure they accurately reflect the required reality. The complete model is presented below.

```
reset;
param n; #number of time periods in my planning horizon
set I; # set of parts
set K; # subset of parts that go through bottleneck
set T:= 1..n;
set T0:= 0..n;
param BOM{I,I} default 0; # BOM bill of materials, how many units of part i are needed for part j
param D{I,T} default 0; # Demand for part i at time t
param R{K}; # processing time of part i in bottleneck
param C; # Capacity per time period
param L{I} default 1; #lead time of part i
param A{I,T} default 0; # available quantity of part i at time t
param P{I}; #penalty for part i
param H{I}; # holding cost of part i

var X{I,T} >=0 integer; /* Production of part i that get started at time t*/
var F{I,T} >=0; /* Fulfilled customer demand of part i at time t*/
var U{I,T}>=0; /* Cumulative unfulfilled demand of part i at time t*/
var V{I,T0}>=0; /* Inventory of part i at time t */


minimize cost: sum{i in I, t in T} (P[i]*U[i,t]+H[i]*V[i,t]);
/* minimize penalty for unfulfilled demand plus inventory costs */

subject to Inventory {i in I, t in T: t>L[i]}:
           V[i,t]=V[i,t-1]+X[i,t-L[i]]-F[i,t]-sum{j in I} BOM[i,j]*X[j,t] ;

subject to InventoryLT {i in I, t in T:  t<= L[i]}:
           V[i,t]=V[i,t-1]+A[i,t]-F[i,t]-sum{j in I} BOM[i,j]*X[j,t] ;

subject to Unfilled {i in I , t in T}:
           U[i,t] = sum{w in T:w<=t} D[i,w] - sum{w in T: w<=t} F[i,w];

subject to Capacity {t in T}: sum{i in K} R[i]*X[i,t] <= C;


subject to InitialInventory {i in I}: V[i,0]=0;

data AssemblyCapacity.dat;
option solver gurobi;
solve;
display X, U, F, V, cost,_total_solve_time;
```

Figure 7. Q4 Assembly Model with Holding Cost and Penalty Cost

For the data file the new sets and the parameters were added as is shown in the following figure.

```
param n:=10;
set I:= 1, 2, A, B, C, D, E, F, G, H, I;
set K:= C, D, E, H, I;

param BOM:=
A 1 4
B 1 2
C A 2
C B 3
D A 1
E B 2
A 2 3
F 2 1
G 2 2
H G 3
I G 2;

param D: 1 2 3 4 5 6 7 8 9 10:=
1   5   5   5   8   8   8   8   8   5   5
2   5   5   5   8   8   8   8   8   5   5
A   0   0   0   0   0   0   0   0   0   0
B   0   0   0   0   0   0   0   0   0   0
C   0   0   20  0   0   0   0   0   0   0
D   0   0   0   0   8   0   0   0   0   0
E   0   0   0   0   0   0   0   0   0   0
F   0   0   0   0   0   0   2   0   0   0
G   0   0   0   0   0   0   0   0   0   0
I   0   0   0   0   0   0   0   0   0   0
H   0   0   0   0   0   0   0   0   0   0
;
#Every Unit takes 30 minutes to produce
param R:=
C   30
D   30
E   30
H   30
I   60
;

param C:= 10080; #capacity constrain

param:  H  P:= #holding and penanty cost, in this case penalty cost is 10 times more than holding
I   2   20
C   4   40
H   5   50
D   2   20
E   2   20
A   15  150
B   15  150
G   20  200
F   50  500
1   100 1000
2   200 2000;
```

Figure 8. Q4 Data set

The results of the suggested planning with the proposed model are presented below. Before delving into the results, it's important to clarify that the objective function has changed. We are now analyzing costs, aiming to minimize the penalty cost for unfulfilled demand plus inventory costs. This shift provides a better perspective for decision-making by management, as they will be evaluated in terms of monetary value rather than just units.

```
:       X    U    F    V
1 0     .    .    .    0
1 1     0    5    0    0
1 2     0   10    0    0
1 3     0   15    0    0
1 4     0   23    0    0
1 5     5   31    0    0
1 6     6   34    5    0
1 7     7   36    6    0
1 8    10   37    7    0
1 9    10   32   10    0
1 10    0   27   10    0
2 0     .    .    .    0
2 1     0    5    0    0
2 2     0   10    0    0
2 3    14   15    0    0
2 4    15    9   14    0
2 5    10    2   15    0
2 6     8    0   10    0
2 7     8    0    8    0
2 8     5    0    8    0
2 9     5    0    5    0
2 10    0    0    5    0
A 0     .    .    .    0
A 1     0    0    0    0
A 2    42    0    0    0
A 3    45    0    0    0
A 4    50    0    0    0
A 5    48    0    0    0
A 6    52    0    0    0
A 7    55    0    0    0
A 8    55    0    0    0
A 9     0    0    0    0
A 10    0    0    0    0
B 0     .    .    .    0
B 1     0    0    0    0
B 2     0    0    0    0
B 3     0    0    0    0
B 4    10    0    0    0
B 5    12    0    0    0
B 6    14    0    0    0
B 7    20    0    0    0
B 8    20    0    0    0
B 9     0    0    0    0
B 10    0    0    0    0
C 0     .    .    .    0
C 1    84    0    0    0
C 2    91    0    0    0
C 3   130   19    1    0
C 4   150   19    0    0
C 5   146    1   18    0
C 6   171    1    0    0
C 7   170    0    1    0
C 8     0    0    0    0
C 9     0    0    0    0
C 10    0    0    0    0
D 0     .    .    .    0
D 1    42    0    0    0
D 2    45    0    0    0
D 3    50    0    0    0
D 4    48    0    0    0
D 5    52    8    0    0
D 6    55    8    0    0
D 7    56    8    0    0
D 8     7    7    1    0
D 9     0    0    7    0
D 10    0    0    0    0
E 0     .    .    .    0
E 1     0    0    0    0
E 2     0    0    0    0
E 3    20    0    0    0
E 4    24    0    0    0
E 5    28    0    0    0
E 6    40    0    0    0
E 7    40    0    0    0
E 8     0    0    0    0
E 9     0    0    0    0
E 10    0    0    0    0
F 0     .    .    .    0
F 1     0    0    0    0
F 2    14    0    0    0
F 3    15    0    0    0
F 4    10    0    0    0
F 5     8    0    0    0
F 6    10    0    0    0
F 7     5    0    2    0
F 8     5    0    0    0
F 9     0    0    0    0
F 10    0    0    0    0
G 0     .    .    .    0
G 1     0    0    0    0
G 2    28    0    0    0
G 3    30    0    0    0
G 4    20    0    0    0
G 5    16    0    0    0
G 6    16    0    0    0
G 7    10    0    0    0
G 8    10    0    0    0
G 9     0    0    0    0
G 10    0    0    0    0
H 0     .    .    .    0
H 1    84    0    0    0
H 2    90    0    0    0
H 3    60    0    0    0
H 4    48    0    0    0
H 5    48    0    0    0
H 6    30    0    0    0
H 7    30    0    0    0
H 8     0    0    0    0
H 9     0    0    0    0
H 10    0    0    0    0
I 0     .    .    .    0
I 1    63    0    0    0
I 2    55    0    0    7
I 3    38    0    0    2
I 4    33    0    0    0
I 5    31    0    0    1
I 6    20    0    0    0
I 7    20    0    0    0
I 8     0    0    0    0
I 9     0    0    0    0
I 10    0    0    0    0
;

cost = 334240
_total_solve_time = 0.181428
```

Figure 9. Q4 Results

When examining the results, it's evident that for engine 1, the unfulfilled demand remains consistently high throughout the process. However, for the other engines or products, the unfulfilled demand is eventually filled at some point during the process. If this were to be the final model recommended for implementation, one aspect to consider would be recommending an inspection or implementing a Six Sigma process to improve the system specifically for engine 1.

Question 5

Change your model to incorporate the setup times and solve again.

In order to change the model and add setup times there are 2 new parameters to add.

1. Parameter S: The setup cost for each item I in the bottleneck
2. Parameter M: Is a large value that will help in the setup constrain.
3. Variable Z: Binary variable that track if a changeover happens at time t.

With the previous things mentioned, the model is shown below.

```
reset;
param n;
set I;
set K;
set T=1..n;
set T0=0..n;
param BOM{I,I} default 0;
param D{I,T} default 0;
param R{K};
param C;
param L{I} default 1;
param A{I,T} default 0;
param P{I};
param H{I};
param S{K};
param M default 9999;

var X{I,T} >=0 integer; /* Production of part i that get started at time t*/
var F{I,T} >=0; /* Fulfilled customer demand of part i at time t*/
var U{I,T}>=0; /* Cumulative unfulfilled demand of part i at time t*/
var V{I,T0}>=0; /* Inventory of part i at time t */
var Z{I,T} >=0 #binary; /* 1 if a changeover happens at time t, 0 otherwise */

minimize cost: sum{i in I, t in T} (P[i]*U[i,t]+H[i]*V[i,t]);
/* minimize penalty for unfulfilled demand plus inventory costs */

subject to Inventory {i in I, t in T: t>L[i]}:
            V[i,t]=V[i,t-1]+X[i,t-L[i]]-F[i,t]-sum{j in I} BOM[i,j]*X[j,t] ;

subject to InventoryLT {i in I, t in T:  t<= L[i]}:
            V[i,t]=V[i,t-1]+A[i,t]-F[i,t]-sum{j in I} BOM[i,j]*X[j,t] ;

subject to Unfilled {i in I , t in T}:
            U[i,t] = sum{w in T:w<=t} D[i,w] - sum{w in T: w<=t} F[i,w];

subject to Capacity {t in T}: sum{i in K} (R[i]*X[i,t]+S[i]*Z[i,t]) <= C;


subject to Setup {i in I, t in T}: X[i,t] <= M*Z[i,t];


subject to InitialInventory {i in I}: V[i,0]=0;
```

Figure 10. Q5 Assembly Model Setup cost

For the constraints the capacity constraint changed, adding the parameter S to count the set-up cost in the model and a Setup constraint was created using the new variable created to monitor the changeovers.

a) Solve first assuming both X (production) and z (setup) are non-negative continuous variables. For this part the variables were changed

```
var X{I,T} >=0; /* Production of part i that get started at time t*/
var F{I,T} >=0; /* Fulfilled customer demand of part i at time t*/
var U{I,T}>=0; /* Cumulative unfulfilled demand of part i at time t*/
var V{I,T0}>=0; /* Inventory of part i at time t */
var Z{I,T} >=0 #binary; /* 1 if a changeover happens at time t, 0 otherwise */
```

Figure 11. Q5 Model Setup variation 1

The complete solution is presented in annex 1 on the annexes section of this report, however the objective function value is 332222.8518 and there are no integer values that don't make sense in this specific problem.

b) Solve requiring z to be binary variables.

```
var X{I,T} >=0; /* Production of part i that get started at time t*/
var F{I,T} >=0; /* Fulfilled customer demand of part i at time t*/
var U{I,T}>=0; /* Cumulative unfulfilled demand of part i at time t*/
var V{I,T0}>=0; /* Inventory of part i at time t */
var Z{I,T} binary; /* 1 if a changeover happens at time t, 0 otherwise */
```

Figure 12. Q5 Model Setup variation 2

The complete solution is presented in annex 2 on annexes section of this report, however the objective function value is 357143.5573 and there are no integer values, that still don't make sense in this specific case.

c) Solve requiring in addition the production X to be integer. Compare the production schedule, cost, and solution time of the three solutions and explain the differences.

```
var X{I,T}>=0 integer; /* Production of part i that get started at time t*/
var F{I,T} >=0; /* Fulfilled customer demand of part i at time t*/
var U{I,T}>=0; /* Cumulative unfulfilled demand of part i at time t*/
var V{I,T0}>=0; /* Inventory of part i at time t */
var Z{I,T} binary; /* 1 if a changeover happens at time t, 0 otherwise */
```

Figure 13. Q5 Model Setup variation 3

The complete solution is presented in annex 2 on the annexes section of this report, however the objective function value is 360526, now the model is giving integer solutions which is the correct thing in this specific case. In the table below there is the comparison of the different OF values obtained in the 3 different variations of the model.

Variation	Objective Value
1	332222.8518
2	357143.5573
3	360526

Table 2. Q5 Cost Comparison

While the values are closely distributed, it's crucial to note that since the variables are continuous, a possible solution might suggest a fraction like 26.2 engines for instance, which is not practically feasible. Thus, ensuring integer values for the variables is imperative. Although this may result in higher costs due to reduced optimization flexibility for the algorithm, it guarantees realistic solutions. For instance, while the cost variation between Solution 1 and Solution 3 is only 8.5%, the robustness of Solution 3, achieved through integer variables, makes it a more viable option.

Regarding the time solutions the table below resume the times depending on the variation.

Variation	Solve Time
1	0.057098
2	0.213374
3	1.16966

Table 3. Q5 Solve Time Comparison

As mentioned previously, its more complicated to the program to find integer solutions, that why it takes more time to solve the integer problem of variation 3 that the non-negative continues problem of variation 1.

Upon reviewing the three solutions presented in each annex, it's worth noting that the assignment schedule is very similar across all of them. However, what differs are the decimals in variation one and variation two, while in variation three, only whole numbers are used. Despite this, by not considering fractions, discrepancies in certain units between each of the models exist. However, as observed in the rest of the comparisons, the numerical difference isn't significant. Yet, when it comes to implementing model robustness, model three stands out as the most coherent, as it produces whole parts rather than fractions.

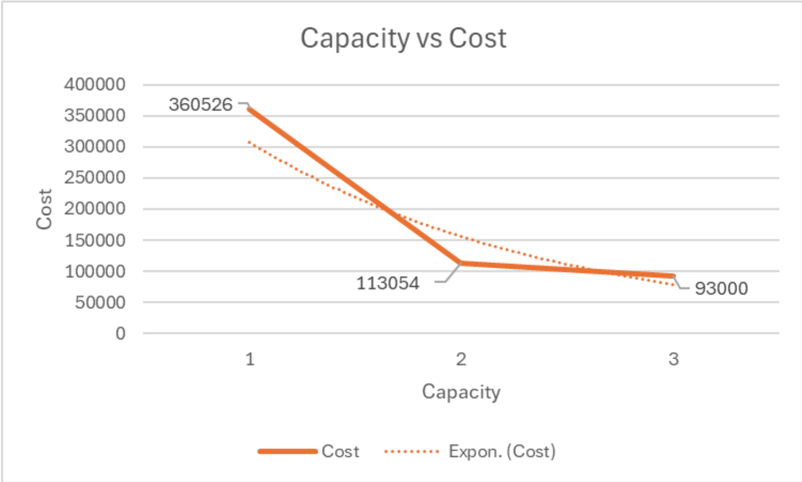
Question 6

In exploring longer term solutions, management would like to understand the value associated with increasing capacity by a factor of 2 or 3? What would you recommend?

With the same model used in question 5C we change the capacity to be the doble and triple and run 3 times the model to se how the cost was going to behave. In the table below are the values used for capacity and a graphic that shows how capacity relates with cost.

Capacity Increase	Capacity Value	Cost
1	10080	360526
2	20160	113054
3	30240	93000

Table 4.. Capacity Values



Graphic 2. Q6 Cost Vs Capacity

In graph two, there's a big drop in costs when doubling capacity. It's really important because the decrease is nearly 220%. In simple terms, when capacity doubles, costs go down to about one-third of what they were. Tripling capacity also lowers costs, but not as much, by 21.6%. So, it's advisable for management to plan to open a new production line to double capacity. Long-term costs will be 220% less than they are now. With this potential saving, equivalent to two-thirds of current costs, there could be a quick or mid-term return on investment, leaning more towards the short term.

References

Muriel, A. (n.d.) *Assembly Systems Questions* [PDF Document].

Annexes

Annex 1 Solution question 5.A

```
ampl: model '/Users/nicoboada/Desktop/Supply Chain Management /Project 2/AssemblySetup.mod';
CPLEX 22.1.1.0: optimal solution; objective 332222.8518
38 dual simplex iterations (0 in phase I)
:        X          U          F          V    :=
1 0      .          .          .          0
1 1      0          5          0          0
1 2      0          10         0          0
1 3      0          15         0          0
1 4      0          23         0          0
1 5      5.02997    31         0          0
1 6      6.90357    33.97      5.02997    0
1 7      6.90357    35.0665    6.90357    0
1 8      10.0397    36.1629    6.90357    0
1 9      10.0397    31.1232    10.0397    0
1 10     0          26.0835    10.0397    0
2 0      .          .          .          0
2 1      0          5          0          0
2 2      0          10         0          0
2 3      14.6039    15         0          0
2 4      14.6039    8.39613    14.6039    0
2 5      9.79226    1.79226    14.6039    0
2 6      8          0          9.79226    0
2 7      8          0          8          0
2 8      5          0          8          0
2 9      5          0          5          0
2 10     0          0          5          0
A 0      .          .          .          0
A 1      0          0          0          0
A 2      43.8116    0          0          0
A 3      43.8116    0          0          0
A 4      49.4967    0          0          0
A 5      51.6143    0          0          0
A 6      51.6143    0          0          0
A 7      55.1588    0          0          0
A 8      55.1588    0          0          0
A 9      0          0          0          0
A 10     0          0          0          0
```

B 0	.	.	.	0
B 1	0	0	0	0
B 2	0	0	0	0
B 3	0	0	0	0
B 4	10.0599	0	0	0
B 5	13.8071	0	0	0
B 6	13.8071	0	0	0
B 7	20.0794	0	0	0
B 8	20.0794	0	0	0
B 9	0	0	0	0
B 10	0	0	0	0
C 0	.	.	.	0
C 1	87.6232	0	0	0
C 2	87.6232	0	0	0
C 3	129.173	20	0	0
C 4	144.65	20	0	0
C 5	144.65	20	0	0
C 6	170.556	20	0	0
C 7	170.556	20	0	0
C 8	20	20	0	0
C 9	0	0	20	0
C 10	0	0	0	0
D 0	.	.	.	0
D 1	43.8116	0	0	0
D 2	43.8116	0	0	0
D 3	49.4967	0	0	0
D 4	51.6143	0	0	0
D 5	51.6143	8	0	0
D 6	55.1588	8	0	0
D 7	55.1588	8	0	0
D 8	8	8	0	0
D 9	0	0	8	0
D 10	0	0	0	0
E 0	.	.	.	0
E 1	0	0	0	0
E 2	0	0	0	0
E 3	20.1199	0	0	0
E 4	27.6143	0	0	0
E 5	27.6143	0	0	0
E 6	40.1588	0	0	0

E 7	40.1588	0	0	0
E 8	0	0	0	0
E 9	0	0	0	0
E 10	0	0	0	0
F 0	.	.	.	0
F 1	0	0	0	0
F 2	14.6039	0	0	0
F 3	14.6039	0	0	0
F 4	9.79226	0	0	0
F 5	8	0	0	0
F 6	10	0	0	0
F 7	5	0	2	0
F 8	5	0	0	0
F 9	0	0	0	0
F 10	0	0	0	0
G 0	.	.	.	0
G 1	0	0	0	0
G 2	29.2077	0	0	0
G 3	29.2077	0	0	0
G 4	19.5845	0	0	0
G 5	16	0	0	0
G 6	16	0	0	0
G 7	10	0	0	0
G 8	10	0	0	0
G 9	0	0	0	0
G 10	0	0	0	0
H 0	.	.	.	0
H 1	87.6232	0	0	0
H 2	87.6232	0	0	0
H 3	58.7536	0	0	0
H 4	48	0	0	0
H 5	48	0	0	0
H 6	30	0	0	0
H 7	30	0	0	0
H 8	0	0	0	0
H 9	0	0	0	0
H 10	0	0	0	0
I 0	.	.	.	0
I 1	58.4155	0	0	0
I 2	58.4155	0	0	0

I 3	39.169	0	0	0
I 4	32	0	0	0
I 5	32	0	0	0
I 6	20	0	0	0
I 7	20	0	0	0
I 8	0	0	0	0
I 9	0	0	0	0
I 10	0	0	0	0

;

_total_solve_time = 0.059225

Annex 2 Solution question 5.B

ampl: model '/Users/nicoboada/Desktop/Supply Chain Management /Project 2/AssemblySetup.mod';
CPLEX 22.1.1.0: optimal integer solution; objective 357143.5573
123 MIP simplex iterations
0 branch-and-bound nodes

:	X	U	F	V :=
1 0	.	.	.	0
1 1	0	5	0	0
1 2	0	10	0	0
1 3	0	15	0	0
1 4	0	23	0	0
1 5	2.68182	31	0	0
1 6	6	36.3182	2.68182	0
1 7	6	38.3182	6	0
1 8	9.13636	40.3182	6	0
1 9	9.13636	36.1818	9.13636	0
1 10	0	32.0455	9.13636	0
2 0	.	.	.	0
2 1	0	5	0	0
2 2	0	10	0	0
2 3	13.913	15	0	0
2 4	13.913	9.08696	13.913	0
2 5	11.1739	3.17391	13.913	0
2 6	8	0	11.1739	0
2 7	8	0	8	0
2 8	5	0	8	0
2 9	5	0	5	0
2 10	0	0	5	0

A 0	.	.	.	0
A 1	0	0	0	0
A 2	41.7391	0	0	0
A 3	41.7391	0	0	0
A 4	44.249	0	0	0
A 5	48	0	0	0
A 6	48	0	0	0
A 7	51.5455	0	0	0
A 8	51.5455	0	0	0
A 9	0	0	0	0
A 10	0	0	0	0
B 0	.	.	.	0
B 1	0	0	0	0
B 2	0	0	0	0
B 3	0	0	0	0
B 4	5.36364	0	0	0
B 5	12	0	0	0
B 6	12	0	0	0
B 7	18.2727	0	0	0
B 8	18.2727	0	0	0
B 9	0	0	0	0
B 10	0	0	0	0
C 0	.	.	.	0
C 1	83.4783	0	0	0
C 2	83.4783	0	0	0
C 3	104.589	20	0	0
C 4	132	20	0	0
C 5	132	20	0	0
C 6	157.909	20	0	0
C 7	157.909	20	0	0
C 8	20	20	0	0
C 9	0	0	20	0
C 10	0	0	0	0
D 0	.	.	.	0
D 1	41.7391	0	0	0
D 2	41.7391	0	0	0
D 3	44.249	0	0	0
D 4	48	0	0	0
D 5	48	8	0	0
D 6	51.5455	8	0	0

D 7	51.5455	8	0	0
D 8	8	8	0	0
D 9	0	0	8	0
D 10	0	0	0	0
E 0	.	.	.	0
E 1	0	0	0	0
E 2	0	0	0	0
E 3	10.7273	0	0	0
E 4	24	0	0	0
E 5	24	0	0	0
E 6	36.5455	0	0	0
E 7	36.5455	0	0	0
E 8	0	0	0	0
E 9	0	0	0	0
E 10	0	0	0	0
F 0	.	.	.	0
F 1	0	0	0	0
F 2	13.913	0	0	0
F 3	13.913	0	0	0
F 4	11.1739	0	0	0
F 5	8	0	0	0
F 6	10	0	0	0
F 7	5	0	2	0
F 8	5	0	0	0
F 9	0	0	0	0
F 10	0	0	0	0
G 0	.	.	.	0
G 1	0	0	0	0
G 2	27.8261	0	0	0
G 3	27.8261	0	0	0
G 4	22.3478	0	0	0
G 5	16	0	0	0
G 6	16	0	0	0
G 7	10	0	0	0
G 8	10	0	0	0
G 9	0	0	0	0
G 10	0	0	0	0
H 0	.	.	.	0
H 1	83.4783	0	0	0
H 2	83.4783	0	0	0

H 3	67.0435	0	0	0
H 4	48	0	0	0
H 5	48	0	0	0
H 6	30	0	0	0
H 7	30	0	0	0
H 8	0	0	0	0
H 9	0	0	0	0
H 10	0	0	0	0
I 0	.	.	.	0
I 1	55.6522	0	0	0
I 2	55.6522	0	0	0
I 3	44.6957	0	0	0
I 4	32	0	0	0
I 5	32	0	0	0
I 6	20	0	0	0
I 7	20	0	0	0
I 8	0	0	0	0
I 9	0	0	0	0
I 10	0	0	0	0

;

_total_solve_time = 0.429634

Annex 3 Solution question 5.C

CPLEX 22.1.1.0: optimal integer solution within mipgap or absmipgap; objective 360526
13178 MIP simplex iterations
3923 branch-and-bound nodes
absmipgap = 32.5007, relmipgap = 9.01479e-05

:	X	U	F	V	:=
1 0	.	.	.	0	
1 1	0	5	0	0	
1 2	0	10	0	0	
1 3	0	15	0	0	
1 4	0	23	0	0	
1 5	2	31	0	0	
1 6	6	37	2	0	
1 7	6	39	6	0	
1 8	9	41	6	0	
1 9	9	37	9	0	
1 10	0	33	9	0	
2 0	.	.	.	0	

2 1	0	5	0	0
2 2	0	10	0	0
2 3	13	15	0	0
2 4	14	10	13	0
2 5	12	4	14	0
2 6	8	0	12	0
2 7	8	0	8	0
2 8	5	0	8	0
2 9	5	0	5	0
2 10	0	0	5	0
A 0	.	.	.	0
A 1	0	0	0	0
A 2	39	0	0	0
A 3	42	0	0	0
A 4	44	0	0	0
A 5	48	0	0	0
A 6	48	0	0	0
A 7	51	0	0	0
A 8	51	0	0	0
A 9	0	0	0	0
A 10	0	0	0	0
B 0	.	.	.	0
B 1	0	0	0	0
B 2	0	0	0	0
B 3	0	0	0	0
B 4	4	0	0	0
B 5	12	0	0	0
B 6	12	0	0	0
B 7	18	0	0	0
B 8	18	0	0	0
B 9	0	0	0	0
B 10	0	0	0	0
C 0	.	.	.	0
C 1	78	0	0	0
C 2	99	0	0	0
C 3	100	5	15	0
C 4	132	5	0	0
C 5	132	5	0	0
C 6	159	5	0	0
C 7	158	2	3	0

C 8	0	0	2	0
C 9	0	0	0	0
C 10	0	0	0	0
D 0	.	.	.	0
D 1	40	0	0	0
D 2	41	0	0	1
D 3	44	0	0	0
D 4	48	0	0	0
D 5	48	8	0	0
D 6	51	8	0	0
D 7	52	8	0	0
D 8	7	7	1	0
D 9	0	0	7	0
D 10	0	0	0	0
E 0	.	.	.	0
E 1	0	0	0	0
E 2	0	0	0	0
E 3	8	0	0	0
E 4	24	0	0	0
E 5	24	0	0	0
E 6	36	0	0	0
E 7	36	0	0	0
E 8	0	0	0	0
E 9	0	0	0	0
E 10	0	0	0	0
F 0	.	.	.	0
F 1	0	0	0	0
F 2	13	0	0	0
F 3	14	0	0	0
F 4	12	0	0	0
F 5	8	0	0	0
F 6	10	0	0	0
F 7	5	0	2	0
F 8	5	0	0	0
F 9	0	0	0	0
F 10	0	0	0	0
G 0	.	.	.	0
G 1	0	0	0	0
G 2	26	0	0	0
G 3	28	0	0	0

G 4	24	0	0	0
G 5	16	0	0	0
G 6	16	0	0	0
G 7	10	0	0	0
G 8	10	0	0	0
G 9	0	0	0	0
G 10	0	0	0	0
H 0	.	.	.	0
H 1	78	0	0	0
H 2	84	0	0	0
H 3	72	0	0	0
H 4	48	0	0	0
H 5	48	0	0	0
H 6	30	0	0	0
H 7	30	0	0	0
H 8	0	0	0	0
H 9	0	0	0	0
H 10	0	0	0	0
I 0	.	.	.	0
I 1	62	0	0	0
I 2	48	0	0	10
I 3	46	0	0	2
I 4	32	0	0	0
I 5	32	0	0	0
I 6	20	0	0	0
I 7	20	0	0	0
I 8	0	0	0	0
I 9	0	0	0	0
I 10	0	0	0	0

;

_total_solve_time = 1.52946