

INDUSTRIAL TECHNOLOGIES PROJECT

Redesign of the FML of the
I4.0 Laboratory, using its
Digital Twin and Simulink



**POLITECNICO
DI MILANO**

Group 134

Cerboni Alessio 1093405

Panti Luca 10707591

Hoffer Oscar 10700936

Rizzuti Elena 10677105

Volpi Eleonora 10674118

SUMMARY

1. PROJECT INTRODUCTION	2
1.1. Introduction and objectives	2
1.2. Description of the AS-IS model.....	2
2. SAMPLING AND DISTRIBUTION ANALYSIS	3
2.1. Sampling	3
2.2. Distribution analysis.....	4
3. THROUGHPUT ANALYSIS	6
3.1. Bottleneck station	6
3.2. Critical WIP	6
3.3. Queue capacity.....	8
4. FML TO-BE	10
4.1. Rebalancing of the line	10
4.2. Variability analysis	11
4.3. Automatization of the manual station	13
4.4. New FML model.....	13

1. PROJECT INTRODUCTION

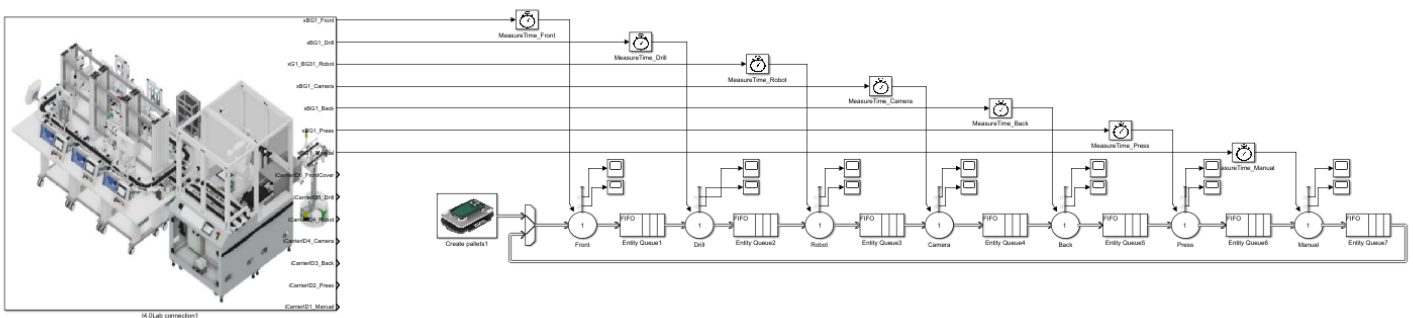
1.1. Introduction and objectives

The aim of this report is to illustrate the analysis and improvement of the DIGITAL TWIN of the Flexible Manufacturing Line (FML), created during the I4.0 Lab at Politecnico di Milano. This model simulates the assembly of a mobile phone with fused PCB, through a sequence of seven stations.

First, we are going to analyse the as-is model, then we are redesigning the line using Simulink, a program in the Matlab environment.

1.2. Description of the AS-IS model

The core centre of the model is the *I4.0Lab* block: it allows to keep track of the signals generated by each of the seven stations in the manufacturing line.



The **Create pallets** represents the entity generator, which generates four pallets. Each of the seven stations is represented by an **Entity Server**, which processes one pallet at time (capacity of one) and whose service time source is the **Signal Port**, which links the station to the I4.0Lab block.

In between each station and the I4.0Lab block there is a **Measure time** block, a clock which returns the activity time of each station.

Conveyors are modelled through **Entity queues**, which work as buffers separating each station with the following one.

Pallets which exit the seventh station go back into the first one, creating a cycle, thanks to the **Entity Input Switch**. This latter block has two inputs ports: the first receives as input the **Create pallets** port, the second is connected to the last station.

Each station has connected two **Scopes**, which allow to visualize the number of entities departed, the number of entities in the block and many other useful values.

The virtual block I4.0 is used to collect the data about the lead time of the stations and the **To Workspace** function is used to load it to Matlab Workspace. To analyse the spread of the service times of each station, the boxplot graphs were drafted. To read the input values for every station from Simulink to Matlab, we used:

2. SAMPLING AND DISTRIBUTION ANALYSIS

2.1. Sampling

We collected lead time data for each of the seven stations given by the I4.0 Lab block for a total running time of 28,800 seconds: 8 hours/shift * 60 minutes* 60 seconds. This long interval of eight hours allows us to collect a sufficient number of measures, in order to make a reliable statistical analysis of the results.

The processing time of each pallet on each single station is equal to the difference between the time of each pallet to exit from the considered machine and the time of the same pallet to entry in the same machine.

The virtual block I4.0 is used to collect the data about the lead time of the stations and the **To Workspace** function is used to load it to Matlab Workspace.

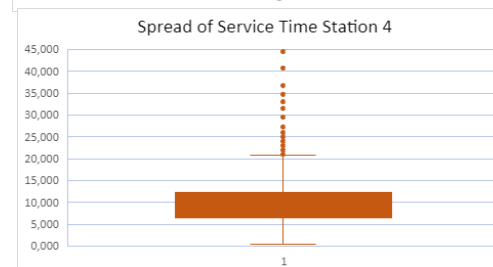
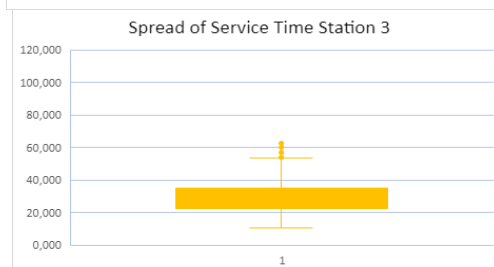
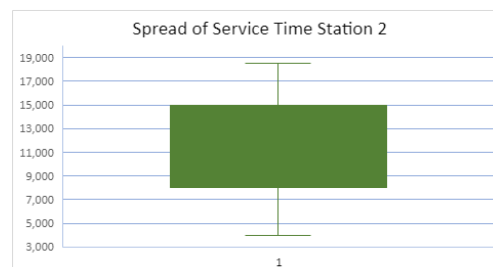
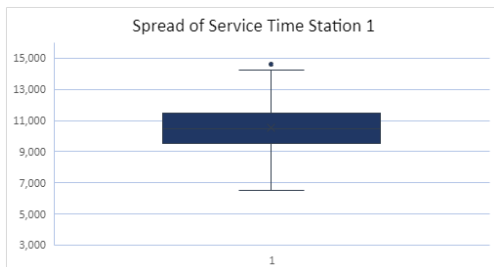
To obtain the data used in the following paragraphs, we run the Matlab code shown in the image beside, applying it to all the seven stations thanks to the Matlab function of the first line. This was done in order to make the code as clear as possible (see Matlab attachments).

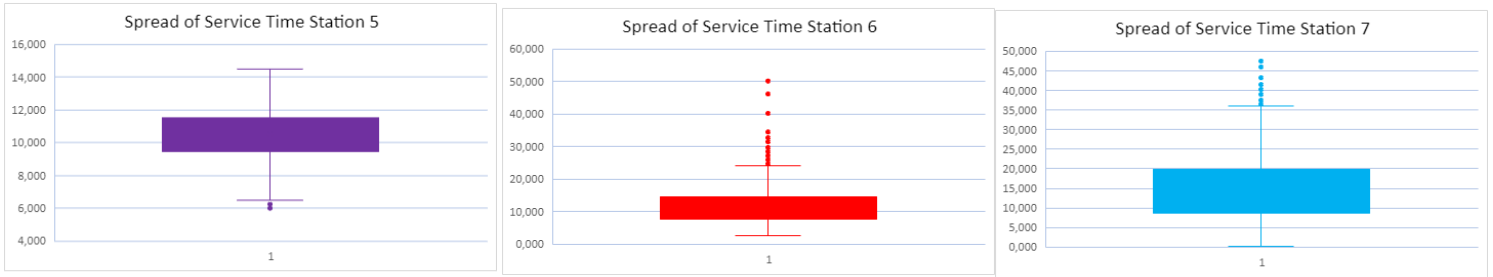
To analyse the spread of the service times of each station, the boxplot graphs were drafted. To read the input values for every station from Simulink to Matlab.

```

1 function [ms,ds,nout,mout,dout,thout,th] = masterfunction(var,j,UR)
2
3
4 data = nonzeros(var); % It eliminates all the initial zeros
5 time = [1:length(data)];
6
7 shortdata = [data(1)];
8 for i = 2:length(data)
9     if data(i-1)~=data(i)
10        shortdata = [shortdata data(i)];
11    else
12        continue
13    end
14 end
15
16 % _____ CALCOLO DELLA MEDIA E DEV.STD. CON OUTLIERS _____ %
17 ms = mean(shortdata);
18 ds = std(shortdata);
19
20 % _____ CALCOLO DELLA MEDIA E DEV.STD. SENZA OUTLIERS _____ %
21 nout = rmoutliers(shortdata);
22 mout = mean(nout);
23 dout = std(nout);
24 % _____ CALCOLO DEL THROUGHPUT RATE _____ %
25
26 thout = 1/mout;
27 th = 1/ms;
28
29 % _____ CALCOLO DEL UR _____ %
30
31 mUR = mean(UR);
32
33 % _____ DISPLAY _____ %
34 disp('_____')
35 fprintf('STAZIONE %d\n',j);
36
37 % fprintf('la media è %f e la deviazione standard è %f\n',ms,ds);
38 fprintf('[senza outliers] la media è %f e la deviazione standard è %f\n',mout,dout);
39 fprintf('[con outliers] la media è %f e la deviazione standard è %f\n',ms,ds);
40 fprintf('Il throughput rate senza outliers (TH) è %f\n',thout);
41 fprintf('Il throughput rate CON outliers (TH) è %f\n',th);
42 fprintf('UR = %f\n',mUR);
43
44 end
45

```

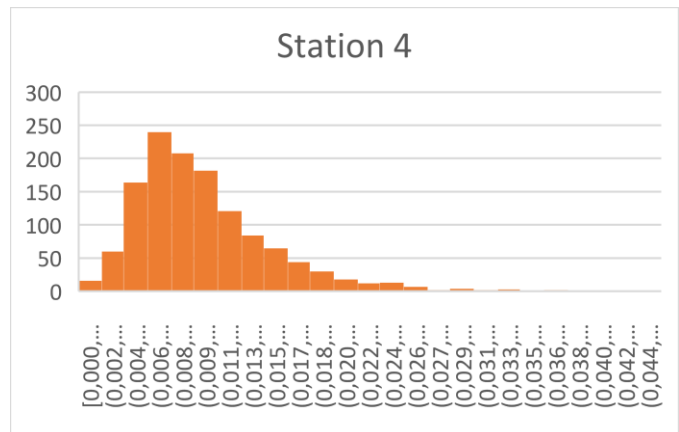
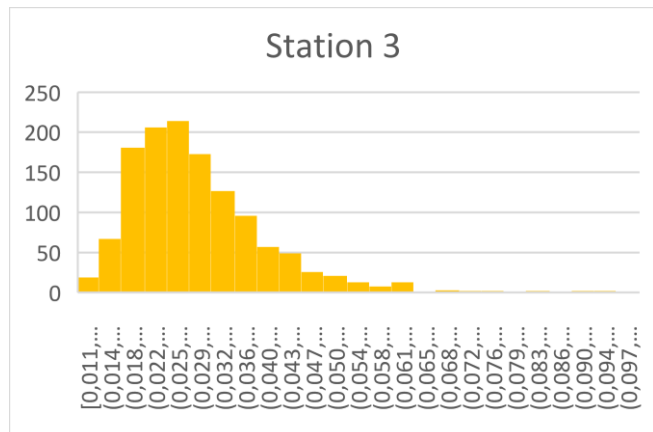
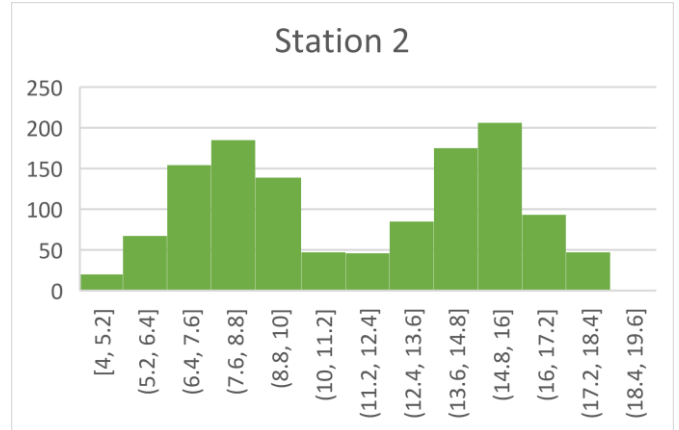
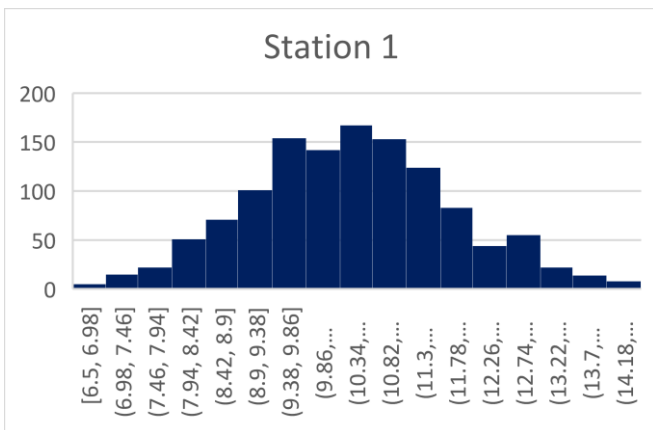


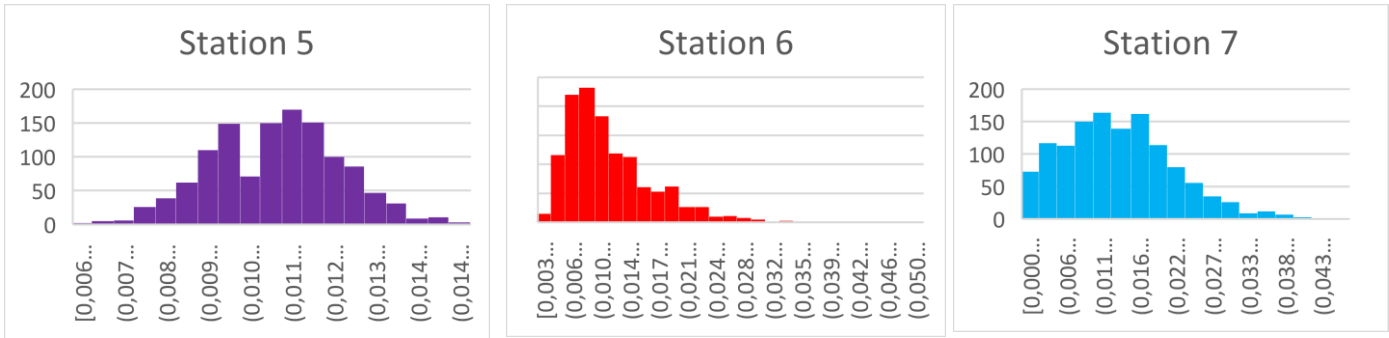


To perform a more precise analysis, we removed the outliers from selected sets of data, using the `remove outliers` command in the Matlab code: `nout = rmoutliers(shortdata)`. Anyway, `rmoutliers` function must be used carefully because it could significantly modify the entire distribution and lead to a distortion of the results. For this reason, by looking at the curves (see the graphs below), we decided not to remove outliers from stations 3, 4 and 6, because the pattern of the curve without high values would not be the same.

2.2. Distribution analysis

We analysed the collected data from the seven stations. By looking at the distribution graphs of the stations, using the `distribution fitter` function on Matlab, we noticed that stations number 1, 2, 5 and 7 have a normal distribution, while stations number 3, 4 and 6 have a lognormal distribution. In fact, when the indicator *likelihood* of the distribution fitter is near 0, it means that the set of data considered has a good fitting with the distribution selected, and it happened for the statistical distributions we decided to associate to each station.





By looking at the graph of machine 2, we identified two normal distributions one after the other: the first one with mean time 8.001s and standard deviation 1.55; the second one with mean time 15.057s and standard deviation 1.361. We used the beside code to simulate the distribution of station 2: the function takes randomly one of the two normal distributions.

```

Main | Event actions | Preemption | Statistics
Capacity:
1
Service time source: MATLAB action
Service time action:
1 x = randsample([0,1],1,true,[0.5,0.5]);
2 if x == 1
3 dt = random('Normal',8.001,1.55);
4 dt = abs(dt);
5 else
6 dt = random('Normal',15.057,1.36135);
7 dt = abs(dt);
8 end

```

Station	1	2	3	4	5	6	7
Population	Normal	Double Normal*	Log* Normal	Log* Normal	Normal	Log* Normal	Normal
μ	10.602	8.001; 15.06	3.3427	2.171	10.6428	2.3739	14.377
σ	1.403	1.55; 1.361	0.3385	0.484	1.3738	0.43	7.8055

* It must be considered that in the case of the Lognormal distribution, μ and σ do not represent the mean and standard deviation, but other parameters specific of this distribution. For what concerns station 2, we included both means and standard deviations of the two normal distributions considered.

From now on, for our analysis, we'll use data without the outlier values for stations 1, 2, 5 and 7, and we considered the complete datasets for stations 3, 4 and 6. In the following table are reported the values provided by the simulation in Matlab, showing the mean time of the lead time and the standard deviation for each station. To measure this data, we computed the difference between the entry and the departure time throughout the simulation and for each station dataset we calculated mean and standard deviation.

Station	Mean time	Standard deviation
1	10.533	1.3985
2	11.5838	3.8163
3	29.4456	10.3844
4	9.6176	5.0776
5	10.5701	1.364
6	11.5233	5.3028
7	14.2009	7.3535

For the computation, in the Entity Generator we defined some attributes to collect the time in which the pallet enters a station and the time in which the service is completed. This has been done through the *ReadTime()* function.

3. THROUGHPUT ANALYSIS

3.1. Bottleneck station

For each of the seven machines, we computed the throughput rate, which is the average output of the production process per unit of time (pallets/seconds).

From these values, the **bottleneck machine** can be clearly identified in station 3, with the lowest throughput, in accord with what emerged from the mean service time computation. This is the station which constrains others to work not at their full capacity.

By the fact that there are many methods to calculate the utilization rate (UR) and to show a more complete situation, we present two different computations of the UR of every machine. The first one is calculated as:

$$UR_i = \frac{TH_{bottleneck}}{TH_i} \text{ with } i = 1, \dots, 7 \text{ station indexes}$$

In this case, it is a matter of fact, depending on the formula, that the UR of the most saturated machine is always 100%. This assumption isn't totally correct and realistic, since UR remains stable even changing the number of WIP in the system.

Anyway, to perform an analysis closer to the real situation, from now on we will focus on the UR taken from the model itself, also showed in the table above, where we rounded up the critical WIP (3.31) to 4 pallets, in order to be able to run the simulation.

Station	Throughput rate	Utilization rate (calculated)	Utilization rate (model)
1	0,09494	35,77%	31,35%
2	0,08633	39,34%	34,19%
3	0,03396	100%	88,87%
4	0,10398	32,66%	29,52%
5	0,09461	35,90%	31,05%
6	0,08678	39,13%	34,97%
7	0,07042	48,23%	42,79%

3.2. Critical WIP

To perform an analysis of the behaviour of the line in terms of throughput's and lead time's trends, we carried out more simulations, varying each time the value of the work-in-process (WIP) pallets. We measured the values of TH and LT for a range of WIP going from 1 to 10, considering it a sufficient value in order to obtain reliable results.

Looking at the TH values, we observed how, by further increasing the WIP value after the fourth simulation, the throughput rate doesn't increase significantly. By definition, the critical WIP is the value from which the TH doesn't improve anymore.

REAL CASE		
WIP	TH(u/s)	mean TTP(s)
1	0,01	99,579
2	0,0184	108,5113
3	0,026	115,593
4	0,0311	128,58
5	0,0332	150,5583
6	0,0338	177,6814
7	0,0338	206,6214
8	0,0339	235,8704
9	0,0339	265,2404
10	0,0339	294,5941

Mathematically, according to the Little's Law, the critical WIP (W_0) corresponds to the better TH achievable by the system (that is station 3's TH=0,03396, since it's the lowest among all stations) times the sum of mean lead time of all the stations in the line:

$$\text{Critical WIP} = r_b * T_0 = 3.31 \text{ units}$$

The following values of TH and TTP were obtained considering the formulas:

- Practical worst case for a given WIP level: $TTP_{PWC} = T_0 + (w-1)/r_b$

$$TH_{PWC} = w * r_b / (W_0 + w - 1)$$

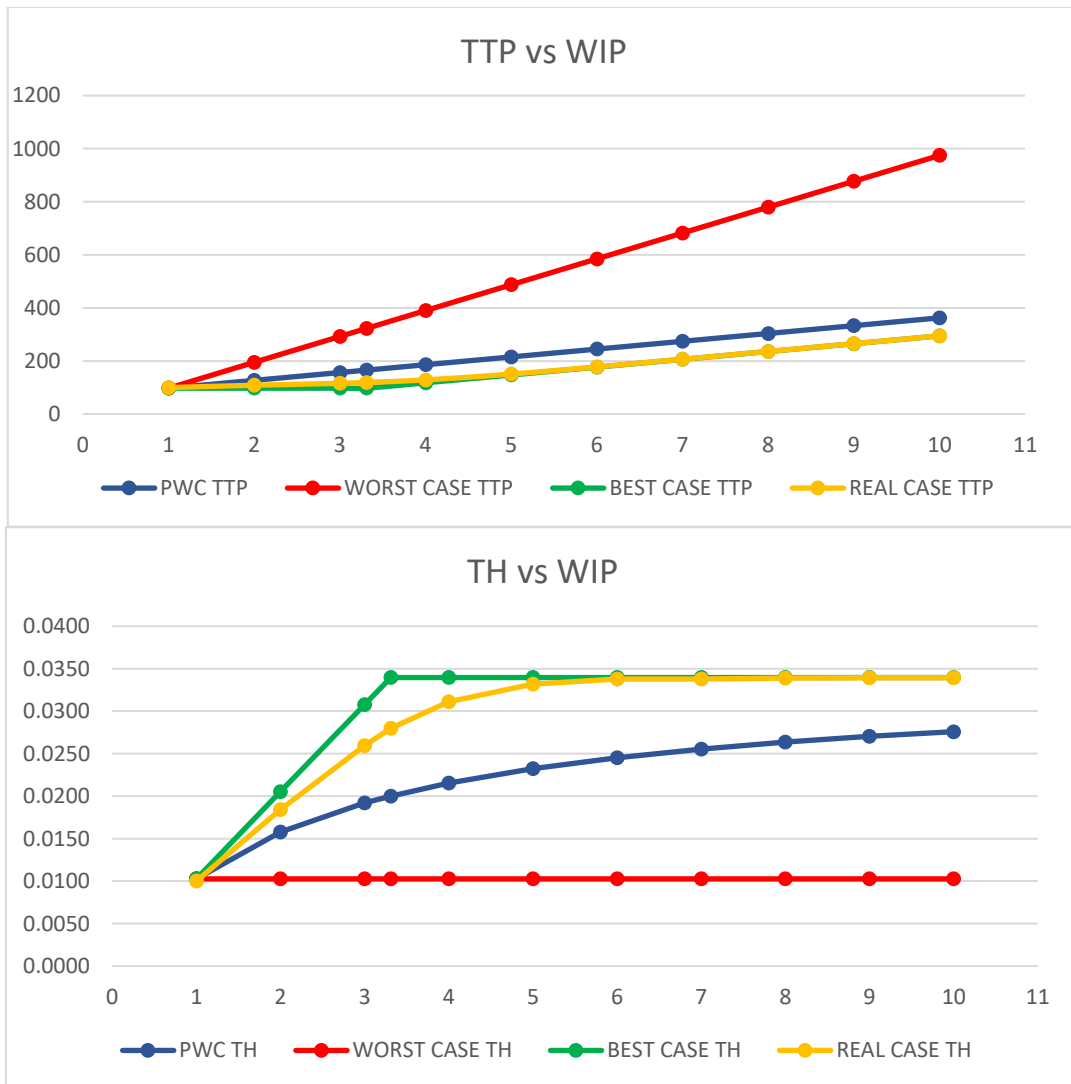
- Worst case for a given WIP level: $TTP_{\text{worst}} = w * T_0$,

$$TH_{\text{worst}} = 1/T_0$$

- Best case given a WIP level: $TTP_{\text{best}} = \begin{cases} T_0 & \text{if } w \leq W_0 \\ \frac{w}{r_b} & \text{otherwise} \end{cases}$, $TH_{\text{best}} = \begin{cases} \frac{w}{T_0} & \text{if } w \leq W_0 \\ r_b & \text{otherwise} \end{cases}$

WIP	PWC TTP	PWC TH	WORST CASE TTP	WORST CASE TH	BEST CASE TTP	BEST CASE TH
1	97,4743	0,0103	97,4743	0,0103	97,4743	0,01026
2	126,9207	0,0158	194,9486	0,0103	97,4743	0,02052
3	156,3671	0,0192	292,4229	0,0103	97,4743	0,03078
3,310	165,5022	0,0200	322,6621	0,0103	97,4743	0,03396
4	185,8135	0,0215	389,8972	0,0103	117,7856	0,03396
5	215,2599	0,0232	487,3715	0,0103	147,2320	0,03396
6	244,7063	0,0245	584,8458	0,0103	176,6784	0,03396
7	274,1527	0,0255	682,3201	0,0103	206,1249	0,03396
8	303,5992	0,0264	779,7944	0,0103	235,5713	0,03396
9	333,0456	0,0270	877,2687	0,0103	265,0177	0,03396
10	362,4920	0,0276	974,7430	0,0103	294,4641	0,03396

Using these values, we were able to confront the curves for the three cases with the real case, drafting the two following graphs: in the first one TTP vs WIP curves are represented, in the second TH vs TTP.

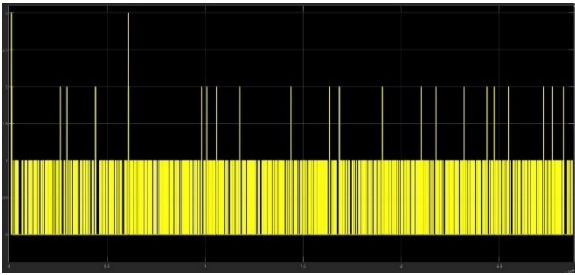


These two graphs show how the real case curves are close to the best-case performances, both for TH and TTP, meaning that the line performs very close to a lean production, with all the stations having a low variability. These overlapping of the curves, especially for high values of WIP, are mainly a result of the simplification assumptions and rounding of the values made during the analysis in the previous paragraphs.

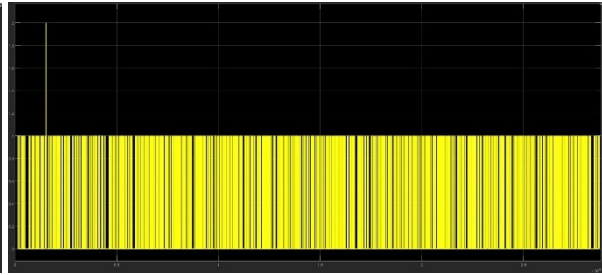
3.3. Queue capacity

To set the capacity of the queues we have looked at the scopes of each of them and analysed the number of entities inside, represented as vertical segments, and in particular the number of pallets that wait before being processed. We have discovered that the maximum number of pallets is always 3 (since we used 4 pallets for the simulation), even if in some stations the frequency of 3 pallets waiting is higher than in other machines.

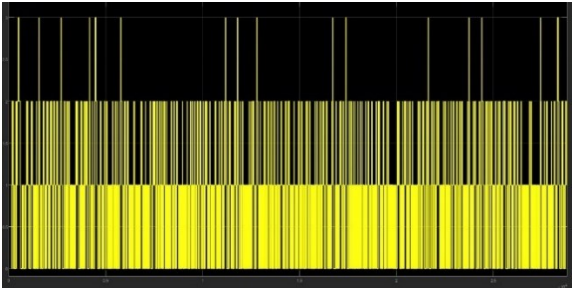
There are stations that mostly need only one or at most two pallets queue, as stations 2,4 and 5. Instead, stations 1,3,6 and 7 require a queue of three pallets. It is clear that station 3 is the most congested, with the highest frequency of 3 pallets waiting.



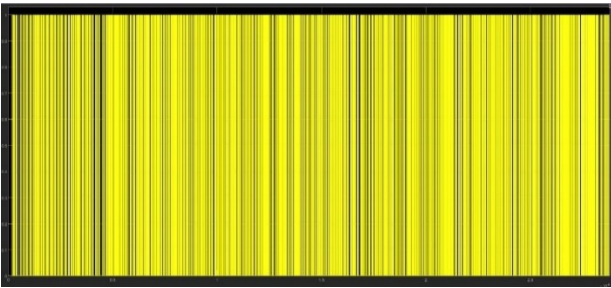
Queue1



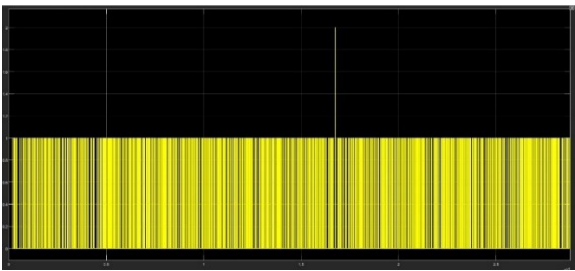
Queue2



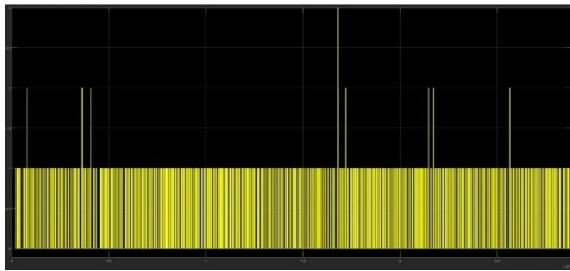
Queue3



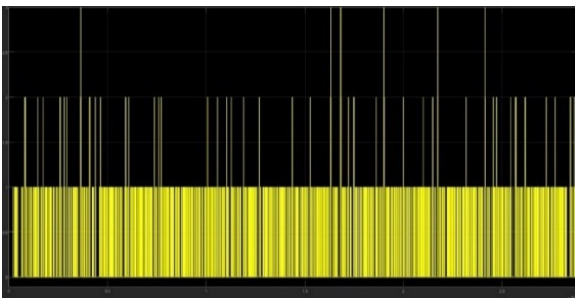
Queue4



Queue5



Queue6



Queue7

In conclusion, the capacity assigned to the queues is the following:

Station	1	2	3	4	5	6	7
Capacity	3	2	3	1	2	3	3

Since the total buffers allocated are seventeen, we managed to respect the constraint of twenty-one imposed by the task.

4. FML TO-BE

4.1. Rebalancing of the line

We are now evaluating different scenarios to understand how to improve and better balance the line. When a production system needs to be improved, it's necessary to focus the attention on the bottleneck, whose slowness affects all the jobs.

The different scenarios we have implemented in Matlab (still using a WIP equal to 4) are explained in the following paragraphs.

First scenario

One machine in every station (As-Is situation): we can observe that station 3 is unquestionably more saturated compared to the others. In fact, while the other machines have a UR which can be considered low, station 3 is the bottleneck because of its high UR of 88.87%.

Station	TH Scenario 1	UR Scenario 1
1	0,0949	31.35%
2	0,0863	34.19%
3	0,0340	88.87%
4	0,1040	29.52%
5	0,0946	31.05%
6	0,0868	34.97%
7	0,0704	42.79%
WIP* =		3,31
mean TTP station 3 =		29,446

Our aim is to arrive to a solution in which all the machines have similar utilization rates, to have a balanced line. The activities can't be merged or separated, so the possible solution is to add new machines to better serve the lowest station. To do it, we have used the Entity switch that allocates two or more machines in parallel. We have changed the setting of the Entity switch from "first port that is not blocked" to "round robin". This way the first pallet waiting in the queue is sent to the first machine, the second to the following machine and so on, starting again from machine 1. Otherwise, the initial setting would have sent the pallet to the first free station starting from machine one, resulting in a UR much higher than the other machines of the station. We have simulated different scenarios with more than one machine per station.

Second scenario

An additional machine is added to station 3, which now has two machines in parallel: although the other stations are more saturated and the TH of station 3 slightly increases, the bottleneck station remains this one.

Station	TH Scenario 2	UR Scenario 2
1	0,0949	36.55%
2	0,0863	39.87%
3	0,0679	53.01%
4	0,1040	34.33%
5	0,0946	36.30%
6	0,0868	40.70%
7	0,0704	49.82%
WIP* =		5,62
mean time station 3 =		14,723

Third scenario

A third machine is added to station 3, which now has three machines in parallel: the station with the lower TH and higher UR is now station 7, while station 3 is no longer the bottleneck.

Fourth scenario

Starting from the third scenario situation, a second machine is added to station 7, while in station 3 there are still three machines, assuming that there are no budget limitations.

The bottleneck station is now the sixth, but with an UR which is just a few percentage points far from other stations' utilization rates. This is the best case among the others, due to the fact that, on one hand, the min TH progressively improves through the different scenarios and reaches now the highest value, and, on the other hand, that the stations have now values of UR comparable with the bottleneck: this means that the line gained more efficiency. The final critical WIP with this configuration raises to 6.1, near two times the one of the initial line.

4.2. Variability analysis

We are now going to identify, among stations 4,5 and 6, the one whose variability has the most relevant impact on the other machines, producing a negative influence on the overall system.

First, we have calculated the coefficients of variation c_e of all the stations with the formula:

$$c_e = \frac{\sigma_a}{t_a} \text{ with } t_a \text{ mean time between arrivals}$$

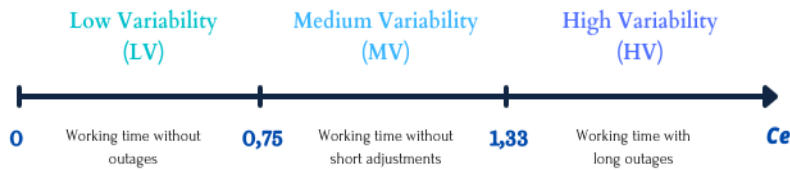
σ_a standard deviation between arrivals

Station	Coefficient of variation
1	0.132
2	0.3295
3	0.3527
4	0.5279
5	0.129
6	0.4602
7	0.544

Station	TH Scenario 3	UR Scenario 3
1	0,0949	36.70%
2	0,0863	40.04%
3	0,1019	36.25%
4	0,1040	34.48%
5	0,0946	36.46%
6	0,0868	40.87%
7	0,0704	50.03%
WIP*=		5,48
mean TTP station 3 =		9,815

Station	TH Scenario 4	UR Scenario 4
1	0,0949	38.58%
2	0,0863	42.09%
3	0,1019	38.01%
4	0,1040	36.21%
5	0,0946	38.33%
6	0,0868	42.93%
7	0,1408	26.92%
WIP*=		6,11
mean TTP station 3 =		9,815
mean TTP station 7 =		7,100

Generally, we can positively observe that all the machines result in the area of low variability. The station with the highest variability among the three is number 4, but not necessarily it has the worst impact over the whole system.



In fact, depending on the level of utilisation of the station, the propagation of variability could be related to the flow of pallets or to the process.

Flow variability out of a low utilization station is determined by the variability of the arrival time of pallets, while with a high utilization station the determinant is the variability within the station and so the processing time, that always has a margin of difference from one piece to the another.

STATION	Mean Time	STD	Coeff. of variation
4	9,87	5,077	0,514387031
5	10,64	1,3738	0,129082572
6	11,83	5,328	0,450380389

In order to perform the sensitivity analysis, we decided to focus on two different cases. Initially, we ran the simulation by considering four pallets circulating in the model. This led to a low UR among the stations, meaning that the overall flow variability is not imposed by the ones of the stations, but by the process.

Afterwards, we carried out the main simulation, introducing 15 WIP in the model, in order to achieve a higher UR and to be able to understand which station has really the most negative impact on the system, when changing its variability.

	TTP			
	Mean	STD	Δ Mean	Δ std
BASE	182,6	28,430		
ST 4	215,34	39,768	32,74	11,338
ST 5	231,27	44,530	48,67	16,100
ST 6	256,72	51,768	74,12	23,338

To change the standard deviation of these stations considered, we have decided to fix the coefficient of variation at 1.5, by the fact that we are making the hypothesis of high variability of the machine considered over the system and observing the one whose impact is more relevant from the three.

The impact on the system was measured taking in consideration the mean and standard deviation of the TTP, to see not only the changing in mean time but also the impact on the TTP's variability. The results show that the worst influence on the overall system is obtained

when modifying station 6's standard deviation, as a proof of that it has the highest delta with the base-case.

Automatization of the manual station

Automating a workstation means installing a robot on the machine, so that the tasks aren't carried out anymore by human operators.

On one hand, robots can work at higher speed and non-stop, without needing breaks or days off, but on the other hand they need periodical maintenance and time to be repaired when breakdowns occur.

As a value added by automation, it becomes easier to collect precise data and monitor lead time and throughput of the stations and the whole system, but also relevant information such as production volume, increase of variability and failures.

Considering the manual machine in station 7, it is convenient to automatize it because it has a high variability: it has a coefficient of variance of 0.544 (higher than all the other machines' values of variation coefficients), meaning that it has a high impact on the line's total variability, mainly because all the other stations are already automated.

Generally, automation could be necessary when certain operations are too dangerous for humans, or the work is so repetitive that it can be easily automated. In this case, in the manual station the operator is simply supposed to unload the finished products from the carrier pallets, so it can be considered a repetitive task that should be automated.

To automatize a station, the initial investment required is high, but it pays off in the long term because productivity increases, and of course variability and frequency of errors decrease, because the operations are not subject to human mistakes anymore.

4.3. New FML model

We have redesigned the model according with the conclusions of the analysis made in the previous paragraphs. The new model is characterized by three machines in station 3, which processes the camera inspection and assemblies PCB and fuses, and two machines in manual station 7, as we established trying to better rebalance the line (see 4.1).

