

# COMBINING SIMULATION AND OPTIMIZATION MODELS ON A PRODUCTION LINE PROBLEM: A CASE STUDY

António A.C. Vieira<sup>(a)</sup>, Edgar Guilherme<sup>(b)</sup>, José A. Oliveira<sup>(a)</sup>, Luís M.S. Dias<sup>(a)</sup> and Guilherme A.B. Pereira<sup>(a)</sup>

<sup>(a),(b)</sup>University of Minho, Campus Gualtar, 4710-057, Braga, Portugal.

<sup>(a),(b)</sup>ALGORITMI Research Center.

<sup>(a)</sup>[antonio.vieira, zan, lsd, gui}@dps.uminho.pt](mailto:{antonio.vieira, zan, lsd, gui}@dps.uminho.pt)

<sup>(b)</sup>[b7542@algoritmi.uminho.pt](mailto:b7542@algoritmi.uminho.pt)

## ABSTRACT

To improve the performance of a production line of a company of the Bosch Group, an optimization model was developed, which produces the optimum allocation of tasks to workstations and workers, according to a set of constraints. These results can thereafter be used in the simulation model, to estimate performance indicators, which would be difficult to estimate with other approaches, namely: waiting times, times spent with displacements and utilization rates. Thus, the purpose of this paper is twofold. First, it describes the combined use of the optimization and the simulation models. Thereafter, it presents the results obtained for 2 scenarios: one without displacements and another with displacements. The former was used to compare the simulation and the optimization models, whilst the later was used to assess the impact of displacements in the production line. By analyzing the results, it was possible to verify that the displacements increased the total time required to produce the devices in more than 10%. Furthermore, it was shown that the displacements caused considerable changes in the remaining performance indicators, indicating the relevance of considering them. This work also brings insights to the Industry 4.0 by proposing an approach to virtualize a production line system, providing the benefits of the 3D visualization of the simulation tool used in this research.

Keywords: Simulation, Optimization, Simio, Production line, Case Study, Industry 4.0.

## 1. INTRODUCTION

In production contexts, production lines are used to produce several types of devices. For this purpose, several workstations are used, on which workers perform different operations. In this regard, the optimization of the logistic movements is crucial for these kinds of problems. For instance, it is important to minimize the occurrence of interception between the trajectories performed by each worker, as they change from one workstation to another.

Assembly line production systems are present in different industrial environments and are used to manufacture a large variety of products. Assembly lines

were developed for a cost efficient mass-production of standardized products to exploit a high specialization of labor and the associated learning effects (Boysen, Fliedner, and Scholl 2006). The assembly lines can be as follows:

- Modular Assembly (Leisner and Ost 1996) – This is an advanced assembly line method that is designed to improve throughput by increasing the efficiency of parallel subassembly lines feeding into the final assembly line. Modular assembly would involve assembling separate modules on their own assembly lines, then joining them together on a final assembly line.
- Cell Manufacturing (Isa and Tsuru 2002) – This method evolved of the increased ability of machines to perform multiple tasks. Cell operators handle three or four tasks, and robots are used for operations like materials handling and welding. Cells of machines can be run by one operator or a multi-person work cell. In these machine cells it is possible to link older machines with newer ones, thus reducing the investment required for new machinery.
- Team Production (Bukchin, Darel, and Rubinovitz 1997) – Team-oriented production is another development in assembly line methods. Workers used to work at one or two-person work stations and perform repetitive tasks. Now teams of workers can follow a job down the assembly line through its final quality checks. This approach has been hailed by supporters as one that creates greater worker involvement in the manufacturing process and knowledge of the system.
- U-Shaped (Aase, Olson, and Schniederjans 2004) – A line may not be the most efficient shape in which to organize an assembly line. On a U-shaped line, or curve, workers are located on the inside of the curve and communication is easier than in a straight line. Assemblers can see each process; what is coming and how fast; and one person can

perform multiple tasks. Also, workstations along the "line" are able to produce multiple product designs simultaneously, making the facility more flexible. Changeovers are easier in a U-shaped line as well and, with better communication between workers, cross-training is also simplified. The benefits of the U-shaped line have served to increase their use widely.

In the assembly line is also important to know the pace of the work parts moving in it and they can be of three main types (Merengo, Nava, and Pozzetti 1999):

- Moving line – a transport system moves, at a constant speed, the units evenly distributed along the line.
- Paced line – the transport system is periodically moving. When a unit arrives at a station, it remains there for a period of time called cycle time and then it is moved to the following station.
- Unpaced line – is equipped with buffers located between stations. In each station the operator takes a unit from the buffer upstream, performs all the required assembly tasks and then moves the unit to the buffer downstream.

In case of a paced assembly line, the station time of every station is limited to the cycle time as a maximum value for each work piece. Since tasks are indivisible work elements, the cycle time can be no smaller. That situation would be ideal, since it presents no idle time between workstations. What usually happens in real problems is the absence of a common cycle time and it is called an unpaced assembly line, i.e., all stations operate at an individual speed, work pieces may have to wait before they can enter the next station and/or stations may get idle when they have to wait for the next work piece. That situation most of the time cannot be completely solved, instead, the optimization method will try to minimize that idle time.

The standard work is a method that defines how operations should be performed at workstations, preventing operators from performing operations randomly. By describing the operations to be performed, operators can become polyvalent because they have access to all the information and can learn to perform new tasks, which guarantees more flexibility of the production system. Basically, standard work consists of three elements (Lopes 2012):

- Takt-Time – is the rate at which products must be made in a process to meet customer demand.
- Work Sequence – is the precise sequence in which an operator performs tasks within takt-time.
- Standard Inventory – is the parts between operations, including units in machines,

required to keep the process operating smoothly.

If applied correctly, the standard work can bring several benefits (Emiliani 2008), such as: the creation of reference points from which is possible to improve continuously, process control, variability reduction, quality and flexibility improvement, stability (i.e. predictable results) and abnormalities predictability.

In light of this problem, a project is being developed at Bosch Car Multimedia Portugal, which consists in optimizing the allocation of an operation to a worker and to a workstation, given a set of requirements and input data, e.g., the duration of each task and number of devices to be produced. Yet, there are other relevant performance metrics, which are difficult to take into consideration by the developed optimization model. Therefore, a simulation model of this problem was also developed, to complement the optimization model with these relevant metrics, i.e., waiting time of devices to produce, per workstation and utilization rates of workers and workstations.

Considering the above exposed, the purpose of this paper is twofold. First, it documents the work conducted to develop both the optimization and the simulation models. The mixed integer linear programming (MILP) model was implemented with AMPL (A Mathematical Programming Language) code, whilst the simulation model was developed in Simio (Dias et al. 2016). Thereafter, it discusses the obtained results emphasizing on the improvement of the production line of the case study. Such work brings insights into the Industry 4.0 agenda (A. A. Vieira et al. 2018; Wang et al. 2016; Uhlemann, Lehmann, and Steinhilper 2017; Longo 2013), namely in the company hosting this research, due to the benefits of the developed solution: virtualization of production line operations in 3D (Turner et al. 2016), testing alternative scenarios, among others. In fact, one of the main contributes of this research is the use of such tool to communicate with customers of the company, while using the tool to illustrate how a given production line will operate. Despite its interest, this topic is out of the scope of this paper.

This paper is organized as follows. Next section presents the MILP optimization model used to improve the performance of the production line, whilst section 3 achieves a similar purpose, for the developed discrete-event simulation model. The results are displayed and analyzed in section 4. Finally, section 5 discusses the main conclusions and some future research directions.

## 2. THE MILP MODEL

The work sequence optimization can be modelled as a Generalized Assignment Problem (GAP). The GAP under study has the following characteristics:

- Mass-production of one homogeneous product given the production process;
- Paced line with fixed Cycle Time (TCT);

- Deterministic operations time;
- Each task is assigned to only one workstation and only one worker;
- Each workstation has a limited number of workers assigned;
- The workers working time is limited by the assembly line CT;
- The tasks are already assigned to the workstations and the objective is to balance the workers working time;

This GAP problem can be defined by a bipartite graph  $G=(T,W,A)$  where  $T$  is a set of tasks with size  $NT$ ,  $W$  a set of workers with size  $NW$  and  $A$  is a set of arcs between tasks and workers. Each task has a processing time and the objective is to balance the work between the workers. The mathematical model for this problem has the following variables, decision variables and constraints:

$$X_{i,k} = \begin{cases} 1, & \text{if task } i \text{ is assigned to worker } k \\ 0, & \text{otherwise} \end{cases}$$

$$C_i = \begin{cases} 1, & \text{if worker changes to task } i + 1 \\ 0, & \text{caso contrário} \end{cases}$$

$WT_k$  – working time of worker  $k$

**Objective function:**

$$\min \sum_i^{NT} C_i$$

**Subject to:**

$$\sum_k^{NW} X_{i,k} = 1, \quad i = 1, \dots, NT \quad (1)$$

$$\sum_i^{NT} T_i * X_{i,k} = WT_k, \quad k = 1, \dots, NW \quad (2)$$

$$WT_k \leq CT \quad (3)$$

$$X_{i,k} \leq X_{i+1,k} + C_i, \quad i = 1, \dots, NT - 1, k = 1, \dots, NW \quad (4)$$

The objective function minimizes the number of change of workers between tasks aiming to balance the workload for workers. The constraint (1) ensures that each task is assigned to only one worker. The worker working time is calculated in the constraint (2). The constraint (3) ensures that the workers working time is not greater than the cycle time. Finally, the constraint (4) calculates the number of changes. The MILP model was implemented in AMPL and our compact formulation was used to solve the instances using the NEOS Server. The GAP mathematical model was developed in AMPL.

### 3. SIMULATION MODELLING

To develop the simulation model of this problem, the data provided to the MILP had to be considered. these data are provided by the Neos server and can thereafter

be used by regular Excel formulas to obtain data according to what is required by the simulation model, regardless of the output of the MILP model. Thereafter, the tool allows these files to be bound to the simulation model, meaning that whenever a new MILP model is run, the new results can be immediately considered by the simulation model.

This section presents the main steps conducted to develop the simulation model used for this problem. This development was divided in two steps: Data preparation and the development of the main simulation model, covered in the second subsection, which uses the data modelling, described in the first subsection.

#### 3.1. Data Preparation

The data were divided in 3 Excel files, each one concerning the following domains: data of workstations, data of the operations to execute and data of resources involved. After incorporating these data, relationships between them need to be modelled in Simio (A. A. C. Vieira et al. 2017; A. Vieira et al. 2016; A. A. C. Vieira et al. 2018). This allows, for instance, to know which operations are performed by which workers on which workstation. To accomplish this, these relationships were modelled as provided by Figure 1.

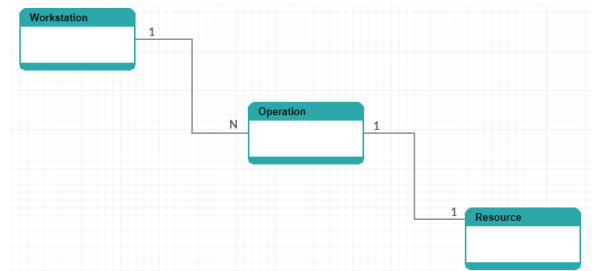


Figure 1: Relationships of data tables.

As the figure suggests, in the problem at hand, at one workstation, there can only be 1 operation at a given time and this operation can only be performed by a single worker. Thus, situation in which, for instance, 2 workers perform the same operation on a given piece, in the same workstation, cannot occur. On the other hand, the same workstation can perform multiple different operations, at different times, therefore there is a one-to-many relationship between the “Workstation” and the “Operation” tables. In this case, the primary key field that identifies a workstation must be incorporated in the “Operation” table, as a foreign key. Lastly, it should be noted that the one-to-one relationship between the “Operation” and the “Resource” tables could be avoided, by merging the content of the two tables in a single one. Yet, this was not adopted, because separating them into 2 tables eased the incorporation of these data into the simulation model. The final of result of incorporating these relationships in Simio is provided in Figure 2, Figure 3 and Figure 4.

servers_distinct	
1	Server1
2	Server1_1
3	Server1_2
4	Server1_3

Data_Table							
	Workstation	Starting Process	Ending Process	Operation	Task Sequence	ProcessingTime (Seconds)	Transport
1	Server1_3	StartingProcess	EndingProcess	operation 36	10		3
2	Server1_3	StartingProcess	EndingProcess	operation 37	20		4
3	Server1_3	StartingProcess	EndingProcess	operation 38	30	1,5	
4	Server1_3	StartingProcess	EndingProcess	operation 39	40		4
5	Server1_3	StartingProcess	EndingProcess	operation 40	50		1
6	Server1_3	StartingProcess	EndingProcess	operation 41	60		2
7	Server1_3	StartingProcess	EndingProcess	operation 42	70		2
8	Server1_3	StartingProcess	EndingProcess	operation 43	80		1
9	Server1_3	StartingProcess	EndingProcess	operation 44	90		3
10	Server1_3	StartingProcess	EndingProcess	operation 45	100		1
11	Server1_3	StartingProcess	EndingProcess	operation 46	110		2
12	Server1_3	StartingProcess	EndingProcess	operation 47	120		2
13	Server1_3	StartingProcess	EndingProcess	operation 48	130		4

5	Server1_4
---	-----------

Figure 2: Data table of workstations.

	Workstation	Starting Process	Ending Process	Operation	Task Sequence	ProcessingTime (Seconds)	Transport
1	Server1	StartingProcess	EndingProcess	operation 1	10		3
2	Server1	StartingProcess	EndingProcess	operation 2	20	1,5	
3	Server1	StartingProcess	EndingProcess	operation 3	30		2
4	Server1	StartingProcess	EndingProcess	operation 4	40		2
5	Server1	StartingProcess	EndingProcess	operation 5	50		2
6	Server1	StartingProcess	EndingProcess	operation 6	60		2
7	Server1	StartingProcess	EndingProcess	operation 7	70	1,5	
8	Server1	StartingProcess	EndingProcess	operation 8	80	7,5	
9	Server1	StartingProcess	EndingProcess	operation 9	90		8
10	Server1	StartingProcess	EndingProcess	operation 10	100		2
11	Server1	StartingProcess	EndingProcess	operation 11	110		1,5
12	Server1	StartingProcess	EndingProcess	operation 12	120		1,5
13	Server1_1	StartingProcess	EndingProcess	operation 13	130		2

Figure 3: Data table of operations.

	Operation	Worker	Reserve	Destination	Reservation Time
1	operation 1	Worker1[1]	True	Input@Server1	Infinity
2	operation 2	Worker1[1]	True	Input@Server1	Infinity
3	operation 3	Worker1[1]	True	Input@Server1	Infinity
4	operation 4	Worker1[1]	True	Input@Server1	Infinity
5	operation 5	Worker1[1]	True	Input@Server1	Infinity
6	operation 6	Worker1[1]	True	Input@Server1	Infinity
7	operation 7	Worker1[1]	True	Input@Server1	Infinity
8	operation 8	Worker1[1]	True	Input@Server1	Infinity
9	operation 9	Worker1[1]	True	Input@Server1	Infinity
10	operation 10	Worker1[1]	True	Input@Server1	Infinity
11	operation 11	Worker1[1]	True	Input@Server1	Infinity
12	operation 12	Worker1[1]	True	Input@Server1	Infinity

Figure 4: Data table of resource allocation settings.

As can be seen, Figure 2 allows to see what operation (from Figure 3) can be performed in which workstation; the server (e.g., Server1 and Server1\_1) object of Simio was used to model workstations. Furthermore, it also allows to see other parameters, such as if the piece is to be transported to the next workstation, the processing time, and others. Lastly, Figure 4 shows the resource parameters required to model other behaviors, such as

to tell the worker if he needs to keep working on the same piece, or if he can proceed to another one – “Reserve” column.

### 3.2. Simulation Model Development

After preparing the data, it was necessary to develop the remaining simulation model. The system in question is comprised by 17 workstations and 8 workers. To develop such a system, 17 Simio Server objects were used and their properties were set as described in Figure 5 and Figure 6. In its turn, to consider 8 workers, it was necessary to place a Worker Simio object and edit a specific property which sets the number of workers. Furthermore, it should be noted that the model is prepared to run the simulations without any type of connection between Servers, by using the Free Space concept of Simio. This was accomplished by modelling a sequence table, which specifies the destination sequence that each entity needs to follow. The last modelling step consisted on setting the properties of the Servers. In this regard, the properties of all Servers were set as indicated in Figure 5.

In most of simulation tools, a process is represented as a random distribution specifying the duration of the process. Yet, in this case, each operation has a processing time associated and several can be performed at a single workstation. Therefore, it was necessary to set the “Process Type” property of the Servers to “Task Sequence”. Thus, it was necessary to specify the operations’ properties, which are specified in two different data tables. Thus, each one was inserted in its corresponding property, i.e., either “Processing Tasks” or “Task Resources”. As Figure 6 suggests, each

property is assigned to a different field of the already presented data tables.

It should be noted that it is only necessary to indicate the name of the column and the data table, because each row of “Operation\_TABLE” is associated to a single operation and the “Process Type” property of all Servers (see Figure 5) is set to “Task Sequence”, i.e., one task per row. The final result of the model while running is illustrated in Figure 7, in 3D. As can be seen, the triangles are the entities of the model, which in their turn represent the pieces of the system which are being produced.

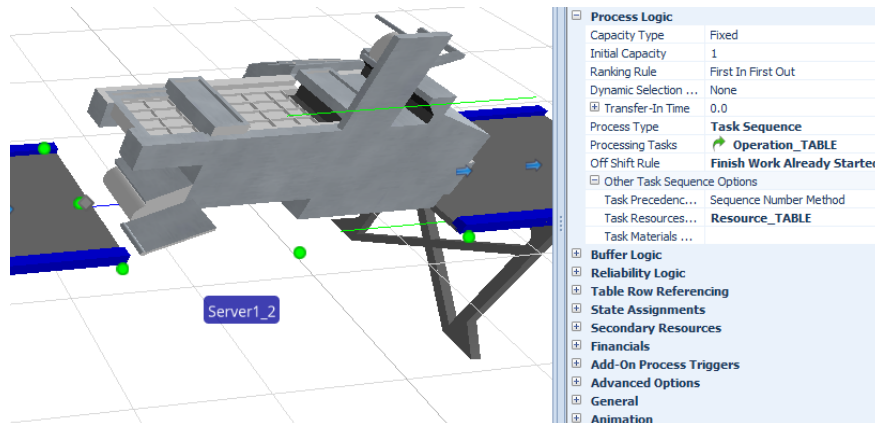


Figure 5: Properties of one workstation of the problem.

<b>Task Information</b>	
Sequence Number	Operation_TABLE.TaskSequence
Name	Operation_TABLE.Operation
Branch Type	Always
Process Type	Specific Time
Processing Time	Operation_TABLE.ProcessingTime
Auto Cancel Trigger	All Immediate Predecessors Cancelled
<b>Resource Requirements</b>	
Object Type	Specific
Object Name	Resource_TABLE.Worker
Selection Goal	Preferred Order
<b>Request Move</b>	
Destination Node	To Node
	Resource_TABLE.Destination
<b>Advanced Options</b>	
Number Of Objects	1
Units Per Object	1
Selection Condition	
Must Simultaneously Seize	False
Keep Reserved If	Resource_TABLE.Reserve
Reservation Timeout	Resource_TABLE.ReservationTime
<b>Material Requirements</b>	
<b>Add-On Process Triggers</b>	
Starting Task	Operation_TABLE.StartingProcess
Finished Task	Operation_TABLE.EndingProcess

Figure 6: Properties that define the task resource settings.

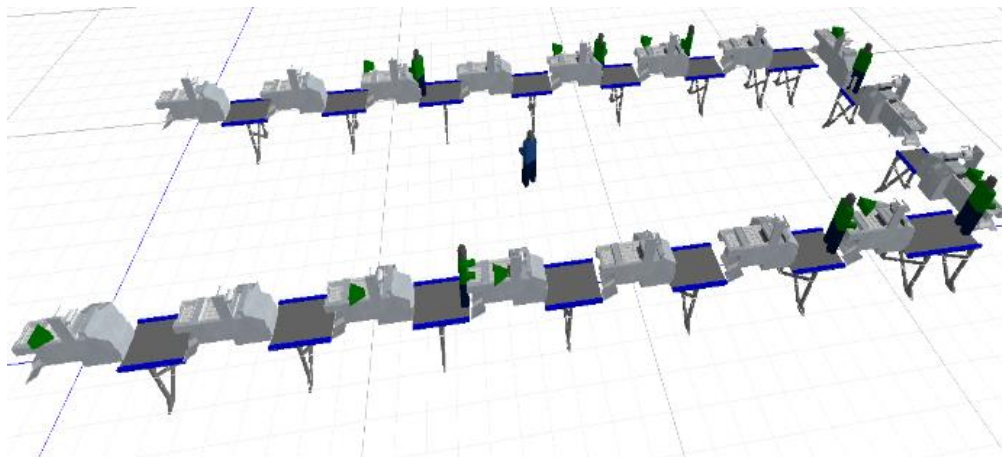


Figure 7: Simulation model running in 3D

#### 4. RESULTS ANALYSIS

This section presents the results of the experiments that were conducted with the purpose of complementing the optimization model with new relevant insights, which are consequence of new Key Performance Indicators (KPI) that the simulation model could obtain, and the optimization model could not. In this regard, 2 experiments were conducted: one which did not consider displacements of workers and materials and a second one which considered them. Whilst the former can be used to compare the performance of both the simulation and the MILP models, the latter is used to obtain the impact of such displacements. The obtained results are displayed in

Table 2, which considered the following KPI:

- KPI1 - Average utilization rates of workstations;
- KPI2 - Average waiting time of devices on each workstation;
- KPI3 - Total simulation time to produce the intended number of devices;
- KPI4 - Average utilization rates of workers;
- KPI5 - Average idle time of workers;

The results of the conducted experiments can be consulted in Table 1 and

Table 2. The former displays the results obtained for KPI1, KPI2 and KPI3. In its turn, the later shows the obtained results for KPI4 and KPI5.

Table 1: Simulation experiments' results for KPI1, KPI2 and KPI3.

Workstation	Scenario I (does not consider displacements)		Scenario II (considers displacements)	
	Utilization Rates (%)	Waiting times (minutes)	Utilization Rates (%)	Waiting times (minutes)
1	92,7	154,6	88,4	174,7
2	31,1	0	26,7	0
3	50,4	0	44,1	0
4	92,2	1,4	87,8	1,6
5	90,8	3,7	88	9,3
6	67,8	0,1	91,8	3,5
7	60,8	0	52,7	0
8	92,4	0,3	85,3	0
9	49,4	0	41,4	0
10	45,1	0	90,7	1,3
11	55,8	0	50,3	0
12	52,8	0	47,5	0
13	77,6	0	71,1	0
14	66,3	0	56	0
15	38,6	0	48,8	0
16	51,9	0	88,2	0
17	90,6	2,1	73	0
Total simulation time (hours)		6,62		7,71

The first thing to notice from the analysis of both scenarios is the simulation time required to run them, i.e., the time required to produce the specified number of device (200 in both cases). As expected, this time was higher in the scenario with the displacements – more than 1 hour of difference, indicating its considerable impact on the overall performance of the production line.

Another interesting aspect to notice from this analysis is that some utilization rates dropped when the displacements were considered (all except workstation 6, 10 and 15). Yet, despite this, in some cases – namely workstation 5, 6 and 10 – the average waiting time for these workstations increased. Furthermore, from this

analysis, it was also possible to obtain the total waiting time on all workstations, which could not be obtained with the optimization model. In fact, it is possible to verify that 162,3 minutes in scenario I and 190,4 minutes in scenario II, divided by all workstation, are spent waiting for a workstation. However, these values include the first workstation, in which devices are queued in the beginning of the simulation. Thus, if this workstation is excluded, the total waiting times drop to 7,7 and 15,7 minutes for scenarios I and II, respectively. Table 2 shows the obtained results from for the KPI 4 and KPI5.

Table 2. Simulation experiments' results for KPI4 and KPI5.

Worker	Scenario I (does not consider displacements)		Scenario II (considers displacements)	
	Utilization Rates (%)	Idle times (minutes)	Utilization Rates (%)	Idle times (minutes)
1	93,3	26,7	89	50,9
2	89,4	42,1	86,8	61,3
3	78,6	85	75,8	112,1
4	80,9	75,8	81,5	85,8
5	91,6	33,5	96,4	16,8
6	97,2	11,2	95,5	20,8
7	91,1	35,3	88,2	54,5
8	92,4	30,1	85,3	67,8

From the analysis of this table, it is possible to verify that - similarly to the results on Table 1 - in the scenario with the displacements, the utilization rate of some workers also decreased, whilst their idle times increased, which seems to reinforce the importance of including the impact of the displacements on this analysis.

Furthermore, the analysis of both tables also suggests that the capacity of some workstations and workers is not being completely used. For instance, the utilization rates of workstation 2 and worker 5 is 26,7% and 96,4%, respectively. On the other hand, this analysis also showed that the impact of the displacements could be of 50% or more. In fact, workstation 10 presents a utilization rate of 45,1% in scenario I, whilst in scenario II this value increases to 90,7. The same happened with workstation 16, which increased its utilization rate from 51,9% to 88,2%.

## 5. CONCLUSIONS

Some of the most relevant Key Performance Indicators (KPIs) for the performance of a production line cannot be estimated with analytical methods. In this regard, a project is being conducted in a company of the Bosch Group, which consisted in optimizing the performance of production lines.

In light of this, a MILP model (Generalized Assignment Problem) is under development. Thus, to complement the MILP model, a simulation model of this problem was also developed in Simio, which applies the object-oriented paradigm and is able to incorporate real industrial data. Furthermore, the simulation tool also offers native 3D visualization, enhancing the communication and involvement of stakeholders, during the project execution, which is aligned with the Industry 4.0 agenda.

In a first instance, the MILP is able to determine the optimum allocation of tasks to a workstation and a worker. The resulting allocation can thereafter be inserted in the simulation model, to properly assess the performance of the production line. To conduct this assessment, in this paper, 2 experiments were considered: one without displacements and a second one which considered these displacements.

The obtained results show the importance of the optimization model to consider such displacements by quantifying the differences between both scenarios. In fact, it could be seen that there was more than 1 hour of difference between the time required to produce 200 devices, by the 2 scenarios. Furthermore, the defined KPI for workstations and workers could vary from one scenario to the other, as much as 50%, demonstrating the need to consider displacements, otherwise misleading conclusions can be withdrawn.

Despite the conclusions obtained from this work, the MILP model still needs to be improved. By improving it, the results obtained from the simulation model will change accordingly, since the simulation model receives an output from the optimization model. Thus, the MILP should be complemented to consider displacements and different layouts. Moreover, other KPI can also be considered, for instance the takt time on each workstation and of each worker.

## ACKNOWLEDGMENTS

This research was partially sponsored by the Portugal Incentive System for Research and Technological Development. Project in co-promotion n° 002814/2015 (iFACTORY 2015-2018) and has been partially supported by FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2019.

## REFERENCES

- Aase, Gerald R., John R. Olson, and Marc J. Schniederjans. 2004. "U-Shaped Assembly Line Layouts and Their Impact on Labor Productivity: An Experimental Study." *European Journal of Operational Research* 156 (3): 698–711. [https://doi.org/10.1016/S0377-2217\(03\)00148-6](https://doi.org/10.1016/S0377-2217(03)00148-6).
- Boysen, Nils, Malte Fliedner, and Armin Scholl. 2006. "A Classification of Assembly Line Balancing Problems." <https://doi.org/10.1016/j.ejor.2006.10.010>.
- Bukchin, Joseph, Ezey Darel, and Jacob Rubinovitz. 1997. "Team-Oriented Assembly System Design: A New Approach." *International Journal of Production Economics* 51 (1–2): 47–57. [https://doi.org/10.1016/S0925-5273\(97\)00060-1](https://doi.org/10.1016/S0925-5273(97)00060-1).

- Dias, Luis M. S., Antonio A. C. Vieira, Guilherme A. B. Pereira, and Jose A. Oliveira. 2016. "Discrete Simulation Software Ranking — A Top List of the Worldwide Most Popular and Used Tools." In *2016 Winter Simulation Conference (WSC)*, 1060–71. IEEE. <https://doi.org/10.1109/WSC.2016.7822165>.
- Emiliani, M.L. 2008. "Standardized Work for Executive Leadership." *Leadership & Organization Development Journal* 29 (1): 24–46. <https://doi.org/10.1108/01437730810845289>.
- Isa, Katsuhide, and Tsuyoshi Tsuru. 2002. "Cell Production and Workplace Innovation in Japan: Toward a New Model for Japanese Manufacturing?" *Industrial Relations* 41 (4): 548–78. <https://doi.org/10.1111/1468-232X.00264>.
- Leisner, Ernst, and Sven Ost. 1996. "Modular Assembly Line System," January.
- Longo, F. 2013. "On the Short Period Production Planning in Industrial Plants: A Real Case Study." *International Journal of Simulation and Process Modelling* 8 (1): 17–28. <https://doi.org/10.1504/IJSPM.2013.055189>.
- Lopes, Sara. 2012. "Aplicação de Standard Work e de Outras Ferramentas de Lean Production Numa Empresa de Elevadores." University of Minho.
- Merengo, C., F. Nava, and A. Pozzetti. 1999. "Balancing and Sequencing Manual Mixed-Model Assembly Lines." *International Journal of Production Research* 37 (12): 2835–60. <https://doi.org/10.1080/002075499190545>.
- Turner, C.J., W. Hutabarat, J. Oyekan, and A. Tiwari. 2016. "Discrete Event Simulation and Virtual Reality Use in Industry: New Opportunities and Future Trends." *IEEE Transactions on Human-Machine Systems* 46 (6): 882–94. <https://doi.org/10.1109/THMS.2016.2596099>.
- Uhlemann, Thomas H.-J., Christian Lehmann, and Rolf Steinhilper. 2017. "The Digital Twin: Realizing the Cyber-Physical Production System for Industry 4.0." *Procedia CIRP* 61: 335–40. <https://doi.org/10.1016/j.procir.2016.11.152>.
- Vieira, A., L.S. Dias, G.B. Pereira, J.A. Oliveira, M.S. Carvalho, and P. Martins. 2016. "Automatic Simulation Models Generation of Warehouses with Milk Runs and Pickers." In *28th European Modeling and Simulation Symposium, EMSS 2016. Winner of the Best Paper Award*, 231–41.
- Vieira, A.A.C., L.M.S. Dias, G.A.B. Pereira, and J.A. Oliveira. 2017. *Agent-Based Simulation to Assess the Performance of Intersections with Pre-Signals: Comparison with Roundabouts. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 10572 LNCS. [https://doi.org/10.1007/978-3-319-68496-3\\_36](https://doi.org/10.1007/978-3-319-68496-3_36).
- Vieira, A A C, L M S Dias, G A B Pereira, J A Oliveira, M Do Sameiro Carvalho, and P Martins. 2018. "Simulation Model Generation for Warehouse Management: Case Study to Test Different Storage Strategies." *International Journal of Simulation and Process Modelling* 13 (4): 324–36. <https://doi.org/10.1504/IJSPM.2018.093761>.
- Vieira, António AC, Luís MS Dias, Maribel Y Santos, Guilherme AB Pereira, and José A Oliveira. 2018. "Setting an Industry 4.0 Research and Development Agenda for Simulation – A Literature Review." *International Journal of Simulation Modelling* 17 (3): 377–90. [https://doi.org/10.2507/IJSIMM17\(3\)429](https://doi.org/10.2507/IJSIMM17(3)429).
- Wang, Shiyong, Jiafu Wan, Daqiang Zhang, Di Li, and Chunhua Zhang. 2016. "Towards Smart Factory for Industry 4.0: A Self-Organized Multi-Agent System with Big Data Based Feedback and Coordination." *Computer Networks* 101 (June): 158–68. <https://doi.org/10.1016/j.comnet.2015.12.017>.