A MONO PARAMETER ANALYSIS ON A SIMULATION MODEL TO SUPPORT GALB HEURISTIC OPTIMIZATION ALGORITHMS BASED ON RESOURCE BALANCING

Sergio A. Gallo[(a)], Giovanni Davoli[(b)], Andrea Govoni[(c)], Francesco Longo[(d)], Riccardo Melloni[(e)], Teresa Murino[(f)]

[(a, b, c, e)]Department of Engineering "Enzo Ferrari", DIEF (ex DIMeC),University of Modena and Reggio Emilia, ITALY
[(d)]DIMEG, University of Calabria, Rende, ITALY
[(a, b, c, e) (c)] Department of Chemical, Material, and Industrial Production, University of Naples – Engineering Faculty– Federico II Napoli, ITALY

[(a, b, c, e)](sgallo, giovanni.davoli, andrea.govoni, riccardo.melloni)@unimore.it, [(d)]francesco.longo@unical.it,
[(f)]teresa.murino@unina.it

ABSTRACT
This paper is based on previous work that presented an approach and the related model to solve a **GALB** problem, and to analyse a multi model manual assembly line, based on **heuristic algorithm** to optimize the scheduling of tasks to the available stations, respecting a **set of restrictions**, as task/station obligation, and **aiming to optimize a multi objective function** based on time and line balancing costs elements.
The assembly line we considered, process a very large variety of items, that differ for size, features, optional, under the increasing market competition.
In that previous work, quite all strategies about resource scheduling opportunities have been considered.
In the present step, we consider the last system we worked on, with a doubled number of stations, to test the reaction to the values changes of some configuration parameter as Due Date, Tardiness Cost, Tardiness Penal, Man work Direct Labour Cost, or Station Saturation.

Keywords: Simulation Models, Decision Support System, Manual Assembly Lines, GALB Problems.

1. INTRODUCTION
The following paper deal with an approach to analyse a multi model manual assembly line, and the following heuristic algorithm.

In particular, at this step we evaluate the reaction of the model as we change values of some configuration parameter.

The scheduling of tasks to the available stations has to respect a set of restrictions, as task/station obligation, and a multi objective function, based on time, balancing utilization rates: Line balancing costs elements can be used as Key Performance Indicator of the balancing performance achieved.

In previous models, Gallo S. A. et al, (2013), in many steps, we developed some model to solve and support the balancing of a manual assembly line, and we start now, from the last development obtained.

So, it could be useful to remind some description.

Tasks assignments to station had to respect efficiency concerns, as the maximization and the balancing of utilization rate, but, moreover, assignment restrictions of specific task to specific station because of the need of special machines, available just at defined station. This constrains are commons for all items of the mix.

The real original considered system we started from in the original and referable configuration, was an assembly line with six stations, and, with six operators. Lot size can vary largely, and it tends to reduce assembly quantity for single order.

Items advancement on line is done on a accumulating conveyor system, so **line** is **paced**, but **not synchronized**.

Assembly line process a very large variety of items, defined in 6 families, as the original Assembly Plan, (**AP**), that differ for size, features, optional, lot size. Very low quantity for single order are commonly accepted.

Task times are stochastic, and, based on real observations, and can be correctly approximated by triangular or lognormal distributions.

The duty of respect the defined external Cycle Time, in a first phase, was not considered, and the algorithms define by itself the final Cycle Time, also if the starting Referable Tack Time was calculated as the maximum value between the Ideal Tack Time, and the maximum Task Time value. Tasks that do not respect line cycle time, just cause the line to increase the performed tack time of the single item in the lot, but not generally of the performed Tack Time mean for the lot.

Just after, when the best configuration has to be defined, the External Cycle Time Is considered as we evaluate the AP processing time, and the final data to be compared with a specific Due Date.

Also in the present model no cost of off line completion was considered, but both Tardiness Daily Cost, and a Lateness Penal Cost are introduced.

A precedence diagram supports technological constrains, very similar among items and, not all operation are performed for all items, depending on item features, and optional.

The same tasks of different items, has operating times, that can vary for each item, for the operation declared in the same way. All tasks in the whole annual assembly mix in AP are represented on the precedence diagram, uniform for all items. To assembly a model not all the 34 operations are needed for a specific item, depending on features and optional. Each item has a defined number of operation which ID number increase as the assembly process goes on.

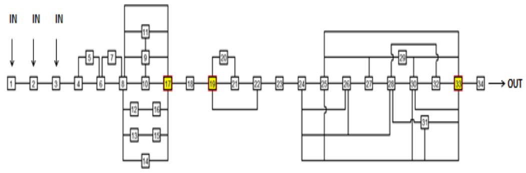

Fig. 1: combinatorial diagram of sequences

At this phase, the model and the system we started from is a U-Shape conveyor, with a doubled number of station compared to the real situation. The idea to double station number, was to enforce balancing opportunities based on the last balancing strategy we applied: the resource/work balance. The constrains position at specific stations has been doubled too, to respect the proportion and homotheticity of the original system.

The performance parameters in the last model were tack time reduction, equal to the production rate maximization, and, on an opposite way, optimize the internal balancing of tasks for stations, or levelling labour level among operators, and they have been considered both in a whole performance or objective function, a cost function.

A virtual model of the assembly line has been built in a simulation environment, to test and measure performances of the heuristic algorithms, where moreover, algorithm code has been implemented in the same software platform to use simulation suite both as a verifying tool, and both as a solving mean, or solution finder, and definitively, as task and resource scheduler.

This suite is Automod®, and the actual number of code lines is quite equal to six thousands.

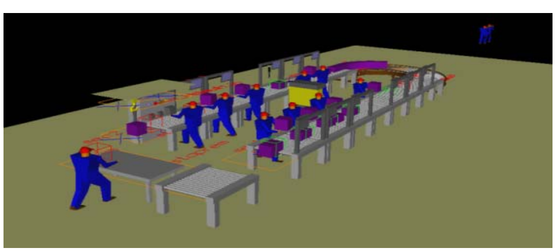

Figure 1: screenshot of the doubled assembly line.

Logics in the heuristic model has been wrote with the aim to be as more general and flexible as possible, to consider the more general problem possible, with easy configurations of data, both processing, both configuration ones.

Simulation models can read AP data from a CSV file, with any useful attribute to be used to characterize the specific Order Line, **OL**, and the configuration: time distribution parameters, item definition, order quantity, order date, etc.. In this way, is easy enough to change configuration.

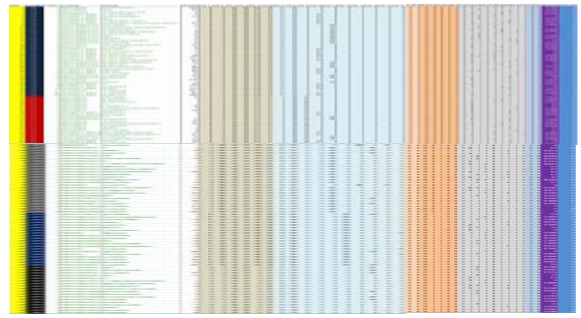

Figure 2: whole mix production plan with parameters values and task times.

We moved from the logic of the last previous model, so, there is a programming code to support all the logics and the heuristic approach to assigning tasks to stations in sequence, attempting to fulfil them, till total station time doesn't overpass calculated the current referable Tack Time. Additional control code check if any constrained task is joined, and in this case, provide to verify the station where to assign that task, if is not the actual one, and to calculate all parameters for intermediate stations (**config_1**).

Uncharged operators are pointed and memorized, and in **config_2**, they are assigned to overcharged station/operators, from the higher level to the lower one.

Just the best allocation will survive, and will be considered for following improving strategies.

Moreover, in **config_3** strategy, if some station is undercharged, based on a defined level, a routine operates a tack time recursive increment till, all the undercharged station become empties, so freed operators could be reassigned to over charged station, as in the previous mentioned case.

In **config_4**, after operators that were “naturally” free, in other words, that were free before increasing the Reference Tack Time, are tried to be assigned to the most overcharged stations, to help already assigned operators.

In **config_5**, also operators freed because of the Reference tack Time increase, are tried to be assigned to the current overcharged stations.

In **config_6**, the best balancing strategy for each line of the Assembly Plan, is choose in the Final configuration. All performance parameters are calculated again.

After the algorithm implementation to the whole assembly mix plan, output performances has been calculated and analyzed, as, particularly, stations utilization coefficients. As many of them showed not very fulfilled values and also the balance was not so good, having to respect constrains in the system, and specificity of and in our data set, not so many other improvement seemed applicable, consequently we considered to compare each station value to others, and to apply an additional balancing strategy based on the resources: heuristic rules to assign to same operators more than one station charge.

In fact, in **config_7**, an aggregation strategy for stations is applied: in case the utilization Coefficient for a specific station was to low compared to a defined acceptance level, a new logic in the code start to calculate the station that define the Current Tack Time, and then try to couple stations with the remaining higher station time with station with the remaining lower value of station time to assign them to the first of scheduled operators assigned to the station couple.

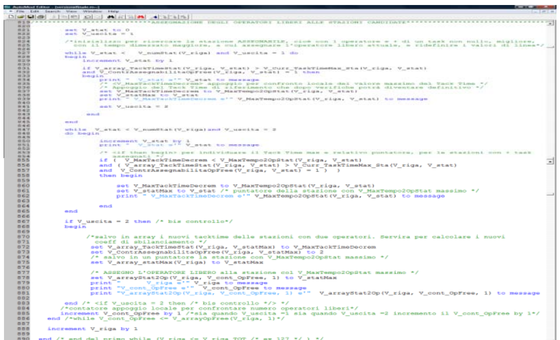

Fig. 3: A snapshot of the logic to assign freed operators and to calculate new RTT.

In this way not all of twelve stations, and related operators, will be scheduled as final optimised configuration. Also in this case, global efficiency is calculated and compared to the previous best one.

For each item just one configuration is the best: for some item it results in six stations and six or more operators, for others, with someone more.

In both strategies all logics to calculate new tack time, to define the station to help are similar, and, once again all the characteristic parameters of each situation will be saved for later comparison with those from previous situations in appropriate variables, with the same name distinct just for the suffix.

In the present work, we evaluate a response analysis when some parameter that we adopted to define thresholds, configurations, acceptance levels, to apply the entire strategy cascade, change values.

Parameters that we considered potentially influent to affect the heuristic algorithm, and that can be used to test the response attitude in the model are:

V_Level_Coeff_UnderUtil that define the percentage of the current reference Tack Time, used to select which station has to be considered undercharged. We considered that the value assumed by this parameter could affect in some case the performance output.

V_accepted_balancing_level, that define which is the percentage level of the Station Time compared to the Current Reference Tack Time that limits the adding of tasks, or stops other improving strategies.

V_DueDate, that defines the last date the assembly Plan can be accomplished and produced.

In fact, in the present work, we consider as a performance issue the respect of a defined Due Date that can help to evaluate in a better way, when is a ameliorative configuration among whole direct cost, balancing level and duration of the production time.

Indeed, in some case, a lower direct cost level can be obtained with lower number of resources but, of course, in some case, because of a better efficiency level, but in other case, because of a lower number of resources, but with an higher production time, also if with a comparable level of efficiency.

With this parameter, a new performance cost function, V_Whole_Cost_xx, is defined, that take in count Direct Cost, but also efficiency and production time, and it is possible discern among solutions in an improved way.

The suffix “xx” stands for any of the specific configuration, the cost function is calculated for.

V_Cost_Modulator, defines the labour direct cost that can shift as better configurations those with better balancing level but with ore operators, toward configurations with a lower balancing level but with less resources.

V_Daily_Tardiness_Cost, that define the value we can adopt to weight the production time and the tardiness. It works for each day, not fractions, that we overpass the Due Date, in a proportional way.

V_Penal defines the cost to pay when due date is overpassed without considering how many days is the tardiness.

Performance parameters of lower level are the production rate, to be maximized, that means to reduce tack time, and, at the same time, optimize the internal balancing of both tasks for stations, both labour level among operators.

Based on these criteria, a multi objective function with the aim to minimize the whole lot assembly cost, calculated on the effective tack time, on the current scheduled resources, on their balancing level, on Penal and Daily tardiness costs, has been defined.

The heuristic algorithm is based, at each step on a logic trying to improve the previous balancing configuration.

The constraint position of a special chamfer machine, allocated on a defined station, and the assignment constrains of other operations, to stations where other equipment is available, dramatically limited opportunities to gain better performances.

2. LITERATURE REVIEW

An assembly line is a flow-oriented production system, where the operative location units performing work, referred to as stations, are sequentially aligned. Work pieces move on transportation systems as a conveyor.

Their configuration and planning is relevant both as a optimization problem both because they are systems at medium intensive capital.

Assembly Line Balancing Problem (**ALBP**) means the assignment of tasks to stations and operators on a line, whereas the items are produced at pre-specified production rate. Configuration planning covers both all tasks allocation and both decisions related to equipping and aligning the productive units for a given production process, including setting the system capacity (cycle time, number of stations, station equipment) as well as assigning the work content to productive units (task assignment, sequence of operations).

Since the times of Henry Ford and the model-T, customer requirements, and consequently, production systems, have changed in a way to increase dramatically customization of their products. The high level of automation of assembly systems and the fixed movement system make the (re)-configuration of an assembly line critical.

In literature, there is a wide variety of algorithms to solve ALBP, any one facing a partial part of the problem, or oriented to a particular system or configuration.

Many of them consider the problem too much statically, just under a one point of view.

But the increasing need to face continuous changes in customer's requirements, as product design, restyling and lot quantity needed, enforced with high customization and reduction of time-to-market, push to test dynamic versions of ALBP solution procedures.

Those modifications imply a very high flexibility level for the line.

ALBP consists of assigning tasks to stations in such a way that (Salveson, 1955):

- each task is assigned to one and only one station;
- the sum of performance task times assigned to each station does not exceed the cycle time;
- the precedence relationships among the tasks are satisfied;
- some performance measures are optimized.

Most procedures consider the types **I and II ALBP**, based on minimization of the number of stations, given a desired cycle time or minimization of the cycle time, given a desired number of stations, respectively.

Because of the simplifying assumptions of this basic problem, this problem was labelled simple assembly line balancing (**SALB**) in the universally accepted review of Baybars (1986). Subsequent works attempted to extend the problem by integrating practice relevant aspects, like U-shaped lines, parallel stations or processing alternatives (Becker and Scholl, 2006), referred to as general assembly line balancing (GALB).

Scholl (1995), and Pierreval et al. (2003) proposed a very large and comprehensive reviews of the approaches developed to solve the problem.

Ghosh and Gagnon (1989) defined a taxonomy to classify ALBP solution procedures under two key aspects, mix or variety of items produced on a single line and the nature of performance task times: single model lines or multi/mixed model lines manufacturing more items in batches or simultaneously; deterministic ALBPs, in with performance task times constant, or stochastic ALBPs, with stochastic task times distributed according to a specific distribution function.

ALBP can be solved to optimize both time - and cost, as reported in Amen (2000, 2001) and Erel and Sarin (1998), which concern the deterministic and stochastic versions of the problem, respectively.

Moodie and Young (1965), Raouf and Tsui (1982), Suresh and Sahu (1994), Suresh et al. (1996) have proposed time-oriented algorithms, improving procedures developed for the single-model deterministic problem, with the aim of minimize stations number and the over time to complete the work off the cycle time.

In any case, relevant incompletion costs often occur in stochastic assembly lines.

A multi objective cost function often is needed.

Two cases, both described in literature:

- the whole line is stopped till the over work is completed (Silverman and Carter, 1986);
- incomplete products get completed off-line.

Kottas and Lau (1973, 1981) proposed heuristic procedures to minimize both the total labour cost and the expected incompletion cost. Extensions of the Kottas and Lau's (1973) method were developed by Vrat and Virani (1976), Shtub (1984).

Sarin et al. (1999) proposed, not so general as Kottas and Lau's (1973), a branch and bound heuristic to minimize the total labour cost and the total expected incompletion cost with good results.

Erel and Sarin (1998) noticed the difficulty of methods in literature to model real conditions, and suggested that newer works should be oriented at useful studies, with impact on real-life assembly lines.

Rekiek (2000) observed that differences among ALBP and real-life statements were the multi-objective nature of the problem, no so considered in literature.

Some studies deal with the re-balancing problem of an existing line, as Sculli (1979, 1984) and, Van Oyen et al. (2001) considered the re-balancing of an existing line, under fluctuations of operator output rates or equipment failures, in short-term problem. The proposed solution to avoid temporary imbalance on the line has been the dynamic work sharing.

Rekiek et al. (2002) demonstrated that the integration between heuristic approaches and multi-attribute decision making techniques is a proven and efficient way for solving assembly lines problems.

The issue of analyse productivity of assembly line through simulation techniques has been faced by De Felice, F. et al (2012) or by Falcone, D. et al (2011), where, through an progressive approach, a simulation model is adopted to produce information about productive capacity, lead time, saturation, Value Added and Non-Value Added activity.

3. SYSTEM CONFIGURATION AND PARAMETERS

Now we start to describe the plan and introduce some parameter for the model.

Model parameters (times in hundredth of minute):

Station Number $k \in [1, n]$ (1)

Task Number $n \in [1, 34]$ (2)

N° of tasks assigned to a single station

$i \in [1, h]$ (3)

Task Time *Top* (4)

Station Time $TStat = \sum_{i=1}^{h} T_{Op}$

$$T_{Stat} = SUC * RTT \tag{5}$$

Operation Unbalancing Coefficient

$$UC_{Op} = \frac{T_{Op}}{Tm} \tag{6}$$

Station Unbalancing Coefficient

$$SUC = \frac{T_{Stat}}{TT_{Line}} \%$$

$$SUC = \sum_i UC_{Op} = \frac{\sum_i T_{Op}}{TT_{Line}} \tag{7}$$

Line Lead Time $LLT = \sum_{i=1}^{k} T_{Stat}$ (8)

Line Tack Time or Cycle Time

$$LTT = Max(\sum_{1=i}^{h} T_{Op}) = Max(T_{Stat}) \tag{9}$$

Mean/Ideal Tack Time

$$ITT = \frac{\sum_{1=1}^{n} T_{Op}}{k} \tag{10}$$

Reference Tack Time

$$RTT = MAX\left(T_{mean} = \frac{\sum_i T_{Op}}{n};\ MAX\left(T_{Op_n}\right)\right) \tag{11}$$

This last parameter, RTT, is the minimum time, in hundreds of minute, used as a limitation roof in the allocation of tasks, maximum threshold the Station Time cannot overcome. It is the maximum value between the ideal tack time (**ITT**), and the value of the larger Task Time (T_{OP}) for each line of the production plan. This value is increased when we are applying the strategy of emptying undercharged stations. RTT represents 100% of work time that can be assigned to each station.

Other parameter, already declared, is SUC, that is the percentage value calculated as the sum of the Unbalancing Coefficient of any operation assigned to the station. It's value is lower or equal of the RTT one, that is the 100%.

We define, at this step, a new cost function, based on the already used Direct Assembly Cost, DAC, but more over on evaluation of Tardiness.

DAC is the product of the Manpower Direct Cost, multiplied by the station number (or operators when more than one is assigned to a station), multiplied by the volume for the OL, multiplied by RTT, to be representative of both the RTT of the line for each row, and for each model, both the number of resources used:

$$Direct\_Assembly\_Cost$$
$$DAC = (RTT \bullet ResNum \bullet LotQuantity \bullet ManWorkCost) \tag{12}$$

Instead, the V_Whole_Cost_xx function is defined in this way:

$V\_Whole\_Cost\_{<config\_number>} = DAC\_{<config\_number>} + V\_Tardiness\_{<config\_number>} * V\_Daily\_Tardiness\_Cost + V\_penal$ (13)

Our assembly line is a multi-mixed model, then we face with a **MALBP** (Mixed-Model Assembly Line BP). We will configure our situation as a **MALB-E** problem, given number of K stations, the aim is to maximize the efficiency E_{line}, i.e. minimizing the Full Cost of assembling the lot.

Our heuristic algorithm is a mix of Work Content and Resource Balancing, that, with the objective function, takes their role and weight, very freely

The simulation code will be used to apply the heuristic logic cascade, to calculate any V_Whole_Cost, distinct for each experiment with specific parameter values. At any step all relevant parameters have been calculated and compared, as in Chutima P. et alter, (2004), Jolai F., et alter, (2008).

4. EXPERIMENTS DESCRIPTION

We decide to define a base configuration with all values for any other parameter set to the middle value.

Values we use are just reasonable, but have to be evaluated, in the future, with a larger extent and ranges, with more values then we did.

Table 1: Values of basic configuration, config_0.

Level Coeff UnderUtil	Accepted Balancing Level	Efficiency Coef	Due Date	Cost Modulator	Daily Tardiness Cost	Penal
0,200	0,800	1,550	8,000	25,000	80000,000	80000,000

All the others values assumed by each parameter can be observed directly on any experiment table, in bold.

We decided to define tree level for each parameter to change with all others values set to the basic configuration.

Values, changed at each experiment, are read on the reading file, "ConfigurationCSV". On this file there are other parameters that we didn't change in this analysis, as the percentage increase value for RTT, the threshold value to divide tasks between two operators when more than one is assigned to a station, and so on.

The strategies to distribute tasks to stations are the same of the previous models.

The approach we adopted has been to define a set of experiments, for all defined set of combination for defined parameters. Then, for each configuration of parameters values, we calculate all of the seven V_Whole_Cost_xx, one value for any of the seven balancing strategy, config_1, config_2, …, cofig_7, that are applied and verified one behind the other in each single run of the simulation of the model.

We have to say that the best approach should be to define a multivariate and multi factor approach, ie to evaluate the effect on a simultaneous variation for all parameters values.

Moreover, the values to use to experiment the model behaviour have to be chosen much more accurately.

But, we are already working on a new version of the model that could do this considering a multi factor and multivariate approach and that could analyse outcome in an automated way.

In this work, we want just test, if the heuristic algorithms keep to perform in a good manner, also when some parameter value change.

Any time a new strategy is applied, called config_xx, all performance parameters, as station/operators utilization coefficients, UCs, Direct Assembly Cost, are calculated, stored and compared to best performing configuration emerged at the previous step for each OL. Just the better, for each item, is the one considered for the final solution.

In last models we operated with defined values for the configuration of the algorithm, so we consider now to evaluate how some variation in the values assumed by some parameter can affect the heuristic performance.

In the following lines, all the experiments configurations.

Table 2: Experiments configuration for "Level Coeff Under" variation: config 1, config 2, config 3.

Level Coeff UnderUtil	Accepted Balancing Level	Efficiency Coef	Due Date	Cost Modulator	Daily Tardiness Cost	Penal	
0,200	0,800	1,550	8,000	25,000	80000,000	80000,000	config 1
0,250	0,800	1,550	8,000	25,000	80000,000	80000,000	config 2
0,300	0,800	1,550	8,000	25,000	80000,000	80000,000	config 3

Table 3: Experiments configuration for "Accepted Balancing Level" variation: config 4, config 5, config 6.

Level Coeff UnderUtil	**Accepted Balancing Level**	Efficiency Coef	Due Date	Cost Modulator	Daily Tardiness Cost	Penal	
0,250	**0,700**	1,550	8,000	25,000	80000,000	80000,000	config 4
0,250	**0,800**	1,550	8,000	25,000	80000,000	80000,000	config 5
0,250	**0,900**	1,550	8,000	25,000	80000,000	80000,000	config 6

Table 4: Experiments configuration for "Due Date" variation: config 7, config 8, config 9.

Level Coeff UnderUtil	Accepted Balancing Level	Efficiency Coef	**Due Date**	Cost Modulator	Daily Tardiness Cost	Penal	
0,250	0,800	1,550	**7,000**	25,000	80000,000	80000,000	config 7
0,250	0,800	1,550	**8,000**	25,000	80000,000	80000,000	config 8
0,250	0,800	1,550	**9,000**	25,000	80000,000	80000,000	config 9

Table 5: Experiments configuration for "Cost Modulator" a Labour Cost parameter: config 10, config 11, config 12.

Level Coeff UnderUtil	Accepted Balancing Level	Efficiency Coef	Due Date	**Cost Modulator**	Daily Tardiness Cost	Penal	
0,250	0,800	1,550	8,000	**10,000**	80000,000	80000,000	config 10
0,250	0,800	1,550	8,000	**25,000**	80000,000	80000,000	config 11
0,250	0,800	1,550	8,000	**30,000**	80000,000	80000,000	config 12

Table 6: Experiments configuration for "Level Coeff Under" variation: config 13, config 24, config 15.

Level Coeff UnderUtil	Accepted Balancing Level	Efficiency Coef	Due Date	Cost Modulator	**Daily Tardiness Cost**	Penal	
0,250	0,800	1,550	8,000	25,000	**40000,000**	80000,000	config 13
0,250	0,800	1,550	8,000	25,000	**80000,000**	80000,000	config 14
0,250	0,800	1,550	8,000	25,000	**120000,000**	80000,000	config 15

Table 7: Experiments configuration for "Level Coeff Under" variation: config 13, config 24, config 15.

Level Coeff UnderUtil	Accepted Balancing Level	Efficiency Coef	Due Date	Cost Modulator	Daily Tardiness Cost	**Penal**	
0,250	0,800	1,550	8,000	25,000	40000,000	**60000,000**	config 16
0,250	0,800	1,550	8,000	25,000	80000,000	**80000,000**	config 17
0,250	0,800	1,550	8,000	25,000	120000,000	**100000,000**	config 18

4.1. ANALYSIS AND COMPARISON OF EXPERIMENTAL RESULTS

In the following lines, we will show the results of the experiments we have outlined, with some consideration.

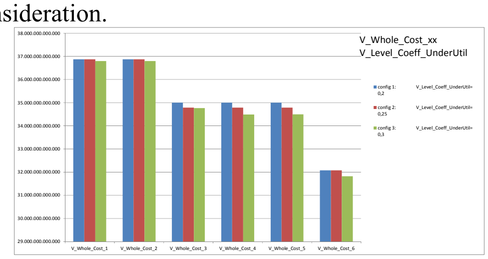

Fig. 4: Whole Cost for config1, config2, config3: Coeff SotUtil variation.

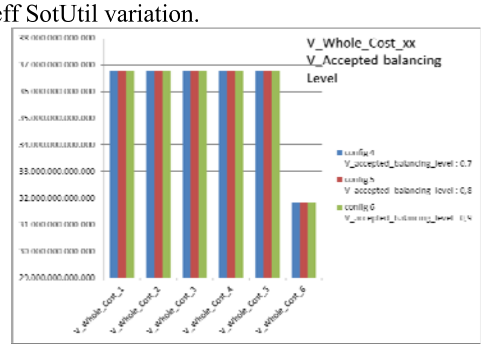

Fig. 5: Whole Cost for config4, config5, config6: V_Accepted balancing Level variation.

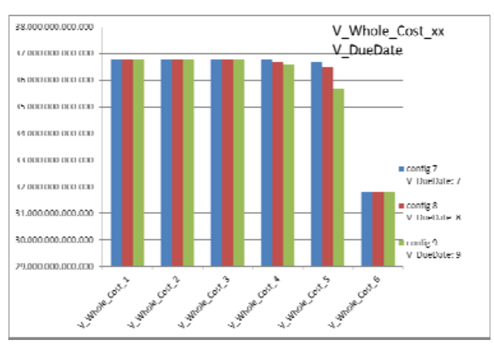

Fig. 6: Whole Cost for config7, config8, config9: Due Date variation.

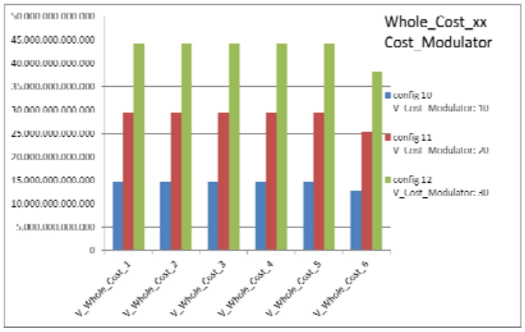

Fig. 7: Whole Cost for config10, config11, config12: Cost Modulator variation.

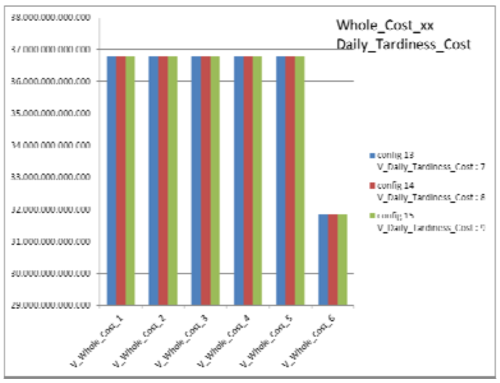

Fig. 8: Whole Cost for config10, config11, config12: Cost Modulator variation.

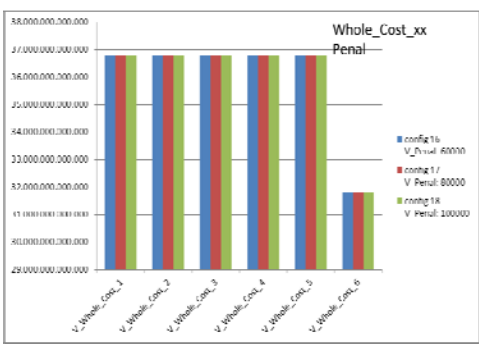

Fig. 9: Whole Cost for config10, config11, config12: Cost Modulator variation.

As we can observe, depending on our set of values, and on the configuration we worked on, some parameter seems to affect more the outline.

In some cases, we can't observe significant differences due to the values we used, this is the case of all the parameter with the exception of the Cost Modulator Parameter that cause a significant difference on the Whole Cost inside each configuration considered, with a reduction for all values just for the aggregation strategy, config_7.

For all others, we can observe that they do not produce relevant differences when different values are used, but much more they affect the effect of increasing of the performance, of the Whole cost function, as we progress to apply any of the defined balancing strategies, but with no significant difference based on values changing. This effect is particularly relevant when considering Level Coef Under Utilization and Due Date, while for others some value effect can be observed, and e global effect when we adopt the strategy to aggregate more than one station to a single operator.

We tried also with other value set, and result, that we do not show here, are conform to those presented.

Finally, we can confirm that the algorithm keep to work appropriately also when some parameter value is changed, because never the new balancing strategy shows a worsening performance compared with the precedent applied. At least, keep the same performance.

We have to remember that the balancing opportunities are strongly affected by constrains position and by task time duration that can give big limitation to accomplish the ideal balancing configuration.

5. CONCLUSIONS

In this paper a developed a new improvement step in the development of an heuristic logic to solve the issue of an manual assembly line.

We tried to evaluate if the logic and the related code keep to perform also when some parameter value change.

The results show that the logic keep to perform and, that confirm the capability of the proposed algorithm of dealing with the multi objective nature of the re-balancing problem.

Now, in this step we can consider solutions that treat with advantages both in tack time reduction, both on balancing and on saturation improvement, but tha can consider also some aspect related to the respect of eventual Due Date.

We are already developing models that can test the model behavior also in case of multi parameter experiment as a deterministic DOE, or multifactorial Analysis, with data set chose to stress the system and to evaluate model limitation.

Obviously, outlines can change in value as you chance parameters values, or if you do not consider that someone exist, but it is important that the approach keep to be valid: any new balancing strategy can just improve the solution, never worse.

Moreover, because we are oriented to the more general solution, in terms of flexibility, variability of mix, number of resources and stations, number and position of some constrains, we are working on optimization approaches that could include a new data collection and the variation of data of the system randomly with logic, to have a greater validation of the algorithm.

REFERENCES

Amen, M., 2000. Heuristic methods for cost-oriented assembly line balancing: *A survey. International Journal of Production Economics 68*, 1–14.

Amen, M., 2001. Heuristic methods for cost-oriented assembly line balancing: A comparison on solution quality and computing time. *International Journal of ProductionEconomics 69*, 255–264.

Bautista J., Pereira J., 2008. "*A Dynamic Programming Based Heuristic for the Assembly Line Balancing Problem*", International Journal of Production Economics.

Baybars, I., 1986. A survey of exact algorithms for the simple assembly line balancing problem. *Management Science 32*, 09–932.

Becker, C., Scholl, A., 2006. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research 168*, 694–715.

Chutima, P. Suphapruksapongse, H., 2004. Practical Assembly-Line Balancing in a Monitor Manufacturing Company, *Tharnmasat Int. J. Sc. Tech.*, Vol. 9, No. 2

De Felice, F. , Petrillo, A., 2012. Productivity analysis through simulation technique to optimize an automated assembly line, Proceedings of the IASTED International Conference on Applied Simulation and Modelling, ASM 2012.

Falcone, D., Silvestri, A. , Di Bona, G., Forcina, A., Pacitto, A., 2011. Study and modelling of very flexible lines through simulation, Proceedings of the 2010 Spring Simulation Multiconference - Emerging M and S Applications in Industry and Academia Symposium, EAIA, Pages 11-16. Boston, MA, USA.

Gallo, S. A., Davoli G., Govoni A., Melloni R., Simulation Models To Support Galb Heuristic Optimization Algorithms Based On Resource Balancing Based On Multi Objective Performance Index, The 25th European Modeling & Simulation Symposium, September, September 25-27, 2013, Athens, Greece.

Ghosh, S., Gagnon, R.J., 1989. A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. *International Journal of Production Research 27*, 637–670.

Erel, E., Sarin, S.C., 1998. A survey of the assembly line balancing procedures. *Production Planning and Control 9*, 414–434.

Gökçen, H K. Ağpak, R. 2006. "Balancing of Parallel Assembly Lines", International Journal of Production Economics.

Kottas, J.F., Lau, H.S., 1973. A cost oriented approach to stochastic line balancing. *AIIE Transactions 5*, 164–171.

Kottas, J.F., Lau, H.S., 1981. A stochastic line balancing procedure. *International Journal of Production Research 19*, 177–193.

Jolai, F., Jahangoshai REZAEE M., Vazifeh, A. 2008. Multi-Criteria Decision Making for Assembly Line Balancing, *Springer Science Business Media*.

Moodie, C.L., Young, H.H., 1965. A heuristic method of assembly line balancing for assumptions of constant or variable work element times. *Journal of Industrial Engineering 16*, 23–29.

Pierreval, H., Caux, C., Paris, J.L., Viguier, F., 2003. Evolutionary approaches to the design and organization of manufacturing systems. *Computers & Industrial Engineering 44*, 339–364.

Raouf, A., Tsui, C.L., 1982. A new method for assembly line balancing having stochastic work elements. *Computers & Industrial Engineering 6*, 131–148.

Rekiek, B., 2000. Design of assembly lines. Memoire presente en vue de l'obtention du grade de docteur en sciences appliquees. *Universite libre de Bruxelles*, Brussels, Belgium.

Rekiek, B., De Lit, P., Delchambre, A., 2002. Hybrid assembly line design and user's preferences. *International Journal of Production Research 40*, 1095–1111.

Salveson, M. E., 1955. The assembly line balancing problem. *Journal of Industrial Engineering* 6, 18–25.

Sarin, S.C., Erel, E., Dar-El, E.M., 1999. A methodology for solving single-model, stochastic assembly line balancing problem. *OMEGA—The International Journal of Management Science 27*, 525–535.

Scholl, A., 1995. Balancing and Sequencing of Assembly Lines. *Physica-Verlag, Heildelberg*.

Scholl, A., Boysen, N., 2009. Designing Parallel Assembly Lines with Split Workplaces: Model and Optimization Procedure. *International Journal of Production Economics*.

Silverman, F.N., Carter, J.C., 1986. A cost-based methodology for stochastic line balancing with intermittent line stoppages. *Management Science 32*, 455–463.

Sculli, D., 1979. Dynamic aspects of line balancing. *OMEGA— The International Journal of Management Science 7*, 557–561.

Sculli, D., 1984. Short term adjustments to production lines. *Computers & Industrial Engineering 8*, 53–63.

Shtub, A., 1984. The effect of incompletion cost on the line balancing with multiple manning of work stations. *International Journal of Production Research 22*, 235–245.

Süer G.A., 1998. Designing Parallel Assembly Lines, *Industrial Engineering Department*, University of Puerto Rico-Mayagüez.

Suresh, G., Sahu, S., 1994. Stochastic assembly line balancing using simulated annealing. *International Journal of Production Research 32*, 1801–1810.

Suresh, G., Vinod, V.V., Sahu, S., 1996. A genetic algorithm for assembly line balancing. *Production Planning & Control 7*, 38–46.

Van Oyen, M.P., Gel, E.S., Hopp, W.J., 2001. Performance opportunity for workforce agility in collaborative and noncollaborative work systems. *IIE Transactions 33*, 761–777.

Vrat, P., Virani, A., 1976. A cost model for optimal mix of balanced stochastic assembly line and the modular assembly system for a customer oriented production system. International Journal of Production Research 14, 445–463.