



(18th ICPR) INTEGRATING INSPECTION POLICY DESIGN IN ASSEMBLY LINE BALANCING

Omar Egidio Carcano, Alberto Portioli-Staudacher

► To cite this version:

Omar Egidio Carcano, Alberto Portioli-Staudacher. (18th ICPR) INTEGRATING INSPECTION POLICY DESIGN IN ASSEMBLY LINE BALANCING. *International Journal of Production Research*, 2006, 44 (18-19), pp.4081-4103. 10.1080/00207540600786681 . hal-00512917

HAL Id: hal-00512917

<https://hal.science/hal-00512917>

Submitted on 1 Sep 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**(18th ICPR) INTEGRATING INSPECTION POLICY DESIGN IN
ASSEMBLY LINE BALANCING**

Journal:	<i>International Journal of Production Research</i>
Manuscript ID:	TPRS-2005-IJPR-0544.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	07-Apr-2006
Complete List of Authors:	Carcano, Omar; Politecnico di Milano, Management, Economics and Industrial Engineering Portioli-Staudacher, Alberto; Politecnico di Milano, Management, Economics and Industrial Engineering
Keywords:	ASSEMBLY LINE BALANCING, INSPECTION, QUALITY MANAGEMENT, LINEAR PROGRAMMING, CONCURRENT ENGINEERING
Keywords (user):	Assembly Line Balancing, Inspection Policy



**INTEGRATING INSPECTION POLICY DESIGN IN ASSEMBLY LINE
BALANCING**

O.E. Carcano, A. Portioli-Staudacher

**Department of Management, Economics and Industrial Engineering, Politecnico di
Milano,**

Piazza Leonardo da Vinci 32, 20133, Milano, Italy

Omar Carcano

Department of Management, Economics and Industrial Engineering, Politecnico di
Milano,

Piazza Leonardo da Vinci 32, 20133, Milano, Italy

e-mail: omar.carcano@polimi.it

phone: +39 02 2399 3952

fax: +39 02 2399 2700

Alberto Portioli-Staudacher

Department of Management, Economics and Industrial Engineering, Politecnico di
Milano,

Piazza Leonardo da Vinci 32, 20133, Milano, Italy

e-mail: alberto.portioli@polimi.it

phone: +39 02 2399 2733

fax: +39 02 2399 2700

ABSTRACT

This paper addresses the problem of assembly line balancing and selection of the best inspection policy. Different inspection policies yield different manufacturing, reworking, inspection and external failure costs. While in literature assembly line balancing and inspection policy design have usually been addressed as different problems, they are strongly interrelated. Inspection tests requires time to be performed and thus their assignation directly influences the results of the assembly line balancing procedure, that is the number of stations that are necessary and hence the line installation cost. Provided what said above, the advantage of integrating inspection policy design in assembly line balancing appears clear, but to our knowledge there is not such a model yet. In this paper a new model that defines simultaneously inspection policy and assembly line balancing is presented.

Keywords:

Assembly Line Balancing, Inspection Policy, Quality Management, Linear Programming, Concurrent Engineering

1. INTRODUCTION

The last years have seen an increasing competition among companies in a context which is changing very rapidly. Quality has progressively become one of the key competitive priorities to compete in the market hence companies have paid more and more attention to this issue.

Quality has a relevant impact on the costs a company has to afford. Indeed not only the external failure cost has to be considered (the cost that a company incurs when a

defective product reaches the market) but also internal quality costs connected to the cost of inspection, manufacturing and reworking. Thus all the above mentioned costs should be considered when defining the inspection policy for a production system.

This paper is focused on assembly line design and aims at integrating inspection policy design into assembly line balancing. In this way indeed it is possible to consider already at a design stage all the costs connected with quality: the external costs, connected to the defective units that reach the customer, and internal costs, i.e the cost of manufacturing, reworking and inspection taking into account the impact that the allocation of inspections has on assembly line balancing. Indeed, as tests requires time to be performed, their allocation strongly influences the line balancing procedure and the results of the balancing procedure itself, that is the total number of workstations that is necessary to set up .

This paper consists of 7 sections. Following the introduction is the review of the background literature in section 2. Section 3 is dedicated to the definition of the problem faced. Section 4 describes the mathematical model, its characteristics, variables, objective function and constraints. In section 5 the design of the simulation experiments that have been run to evaluate the performance of the method proposed is explained. The results of those experiments have been reported in section 6. At the end some conclusions are drawn. An appendix with some details follows up.

2. LITERARY BACKGROUND

2.1 Inspection allocation in serial multistage production systems

Lindsay and Bishop (1964) were the first to formulate the problem of inspection allocation in serial production systems. The problem is solved minimizing an objective function which accounts for inspection, lost production and quality loss costs. Every

step of the production system has a known probability of generating a defect and the quality tests are available after each processing stage. The defective parts are removed from the production system.

Pruzan and Jackson (1967), besides the variable inspection cost, introduced the fixed cost as quota of the amortisation and they made the assignation of tests depend upon the history of the piece, that is the sequence of operations previously assigned. Eppen and Hurst (1974) proposed the first model that considers two types of inspection errors: acceptance of defective units and rejection of conforming parts.

The analysis of inspection allocation in multistage production system has been gradually refined and several contributions on this issue were proposed loosening some of the original hypothesis and proposing different solution methods.

Ballou and Pazer (1982) studied the impact of inspection errors on the final solution and demonstrated that type I errors (rejection of conforming parts) have a greater effect than type two errors (acceptance of defective units). Lubicz (1983) developed the first model to account for non-conformity costs that vary according to the stage where the non-conformity has appeared. Yum and Mc Dowell (1987) proposed a model that permits to consider different management policies for defective units that can be either repaired, replaced, reworked or discarded with costs that vary according to the stage in the manufacturing process. Considering that multiple or repeated inspections can be a way to limit the effects of inspection errors, Raz and Kaspi (1991) proposed a non-linear integer programming model for an integrated approach to sequencing and locating multiple inspection operations which distinguishes itself for the presence of a transfer function that links incoming and outgoing level of quality and costs for each testing and manufacturing operation. Lee and Unnikrishnan (1998) proposed a model that considers the presence of several parts, each one having a different technological cycle defined through the sequence according to which they have been worked on every machine.

This model is characterized by the fact that a constraint on the maximum time for the availability of inspection machines is set. Shiau (2002) formulated a model based on the finite inspection resource constraint (as Lee and Unnikrishnan) and an inspection error model: the inspection error is not constant or a specified probability, but accounts for inspection capability, manufacturing capability and tolerance. Rau and Chu (2005) presented an heuristic to solve the inspection allocation planning problem with two types of workstation, workstation of attributable data (WAD) and workstation of variable data; in addition this study considers three possibilities for the treatment of detected nonconforming units, namely repair, rework and scrap.

2.2 Inspection allocation in non-serial multistage production systems

Britney (1972) was the first one to consider non-serial systems: in his model, that finds the best solution minimising the internal and external cost of quality and the inspection cost, every operation has a defined probability of generating a defective unit, a perfect test is available after every operation and reparation is perfect as well.

Following a path similar to the one described for serial production systems, the models proposed for inspection allocation in non-serial multistage production systems have been gradually refined.

Yum and Mc Dowell (1981) developed Brytney’s model by introducing first type and second type inspection error. Garzia-Diaz, Foster and Bonyquet (1984), taking into account the reduction in inspection error thanks to repeated inspection, developed a dynamic programming model having as variables the position of inspections and the number of repetitions (defined as inspection degree). Also for non-serial system, Kaspi and Raz (1994) considered the problem of allocating both manufacturing and inspection operations whose relative job order is not assigned beforehand but defined through precedence relations defined in an assembly diagram. The probabilities to generate

defective units are known but depend upon the specific operation and the operations previously assigned; I type and II type inspection error are considered and defective parts are discarded.

2.3 Assembly line balancing

Assembly line balancing is a very lively research field. The fundamental line balancing problem is to assign the tasks to assemble a product to an ordered sequence of stations, such that the precedence relations are satisfied and some measure of effectiveness is optimised (e.g. minimize the balance delay or minimise the number of workstations), (Gosh and Gagnon, 1989).

Salveson (1955) was the first one to give a mathematical form to the problem and proposed a linear integer programming model to solve it.

Since then a lot of articles have been published detailing the advances in this field and periodically articles reviewing the models proposed are published .

According to Gosh and Gagnon (1989) the Assembly Line Balancing problem (ALB) and the accompanying research and literature can be classified into four categories: Single Model Deterministic (SMD), Single Model Stochastic (SMS), Multi/Mixed Model Deterministic (MMD), and Multi/Mixed Model Stochastic (MMS). The SMD version of the ALB problem assumes dedicated single-model assembly line where task times are known deterministically, the SMS model introduces the concept of task time variability. The MMD problem formulation assumes deterministic task times, but introduces the concept of an assembly line producing multiple products. The MMS problem perspective differs from its MMD counterpart in that stochastic times are allowed. For each category they distinguish between Simple Case and General Case. The General Case refers to situations in which restrictions or other factors are introduced into the model (e.g. parallel stations, zoning restrictions etc.).

Rekiek, Dolgui, Delchambre and Bratcu (2002) distinguish between simple assembly line balancing and general assembly line balancing problems. Within simple assembly line balancing they classify models considering optimization methods (linear programming, dynamic programming, branch and bound etc.), processing time (deterministic time, stochastic time, dynamic time...), equipment selection, additional constraints (operators, buffers etc.). As to the general problem, they classify methods considering line configuration (U-Line, parallel stations, parallel lines...), design objectives, use of simulation.

We-Min Chow (1990), Scholl (1995), and Nof, Wilhelm and Warnecke (1997) are good references for deepening the knowledge of alternative line balancing methods.

The most recent reviews are the ones by Scholl and Becker (2006) and Becker and Scholl (2006). The former is a review of methods to solve simple assembly line balancing problem and the latter is focused on the generalized assembly line balancing problem.

3. PROBLEM FORMULATION

An assembly line consists of a finite set of tasks characterised by a processing time and precedence relationships that specify the permissible ordering of the tasks. The balancing line problem aims at assigning the tasks necessary to assemble a product to workstations so to minimize a certain objective function, usually the total number of stations and thus the total investment and running cost.

Defining the inspection policy consists in deciding which tasks to control and where to control them.

Several costs are connected to the inspection policy definition: manufacturing cost, reworking cost, inspection cost and external failure cost. As an item is subject to different operations on an assembly line, it can incur defects as a result of processing. If no tests to detect defective operations are allocated, these parts will continue their

working cycle on the line nourishing manufacturing costs; furthermore if the items eventually reach the customer, additional failure cost has to be considered. On one hand if inspections were allocated after every assembly task, defective units would be detected as soon as they are generated reducing manufacturing and reworking cost; on the other, tests imply additional costs for inspection equipment and, as they require time to be performed, they impact on the number of stations that is necessary to set up and thus on the line installation cost.

Besides a processing time, tests are characterised by precedence relationships, exclusion relationships and position added costs.

Precedence relationships take into account the fact that in order to make a test, the performing of some other tasks might be necessary.

Exclusion relationships account for the fact that a test can be no more available after some other tasks have been performed. For example it might be no more possible to check if a component or a group of components work well after some other parts covering the components to be tested have been assembled.

The position added cost takes into account the relative position of a test and the tasks it checks.

Indeed the cost connected to repairing a certain task can increase if the task is not checked, and thus repaired, straight after it is performed. Indeed having performed some assembly tasks between a task and the test that checks this task can imply extra reparation cost if the task is found to be defective by the inspection. This extra reparation cost can be due to extra-disassembly and re-assembly activities or material loss (for example gases and fluids that are used to fill a circuit).

Thus the definition of the inspection policy influences several costs (manufacturing cost, reworking cost, external failure cost, position added cost) as well as the line installation cost.

Nevertheless, while assembly line balancing and inspection policy design are strongly interrelated, to our knowledge, there are no models in literature addressing at the same time these two issues.

The models focused on assembly line balancing do not consider the impact of inspection policy.

Models that consider inspection allocation in multi-stage production system neglect the distinctive characteristics that an assembly line problem presents: the cycle time (constraining the amount of work that can be allocated to a station), the precedence relationships for tasks and tests and the exclusion relationships for tests. Indeed in an assembly line problem, tests are first characterised by a processing time as the time required to perform a test directly influences the number of workstations that is necessary to set up. Furthermore in these models, a test controlling all the tasks previously assigned is available after every production stage, thus the precedence and exclusion relationship cannot be considered as well as the position added cost.

To conclude, existing models considering balancing problems do not consider the inspection policy. The models allocating tests in multistage production systems, even though offer some important references for inspection positioning, neglect some factors that are fundamental for balancing purposes.

Hence the need for a model that allows to define the inspection policy while balancing an assembly line tackling specifically all the issues connected to a balancing problem, appears clear. Several works show that the integrated and simultaneous approach gives relevant benefits when it integrates different functional areas, for instance manufacturing and quality or design and quality (Portioli-Staudacher, 1999; Portioli-Staudacher, Landeghem, Mappelli, Redaelli, 2003).

4. THE MODEL

This paper presents a model to balance an assembly line concurrently assigning inspection tests.

The assembly procedure (with the precedence relationships) is specified in an assembly graph (see figure 1). A set of in-line tests are available, each test controls one task or a set of tasks and is characterised by a processing time τ_k , precedence relationships, exclusion relationships and position added cost. A final test is also available to check all assembly tasks. The final test is done out of the line. Whenever a defective product is detected, it is repaired or substituted. The defective one is repaired offline and then ready to substitute the next defective one. In this case it is necessary to afford reparation/replacement costs and then the product continues the assembly cycle along the line. Perfect reparability of the product is assumed.

The model proposed permits to balance the line to manufacture the product concurrently defining which tests to use and the position where they have to be performed.

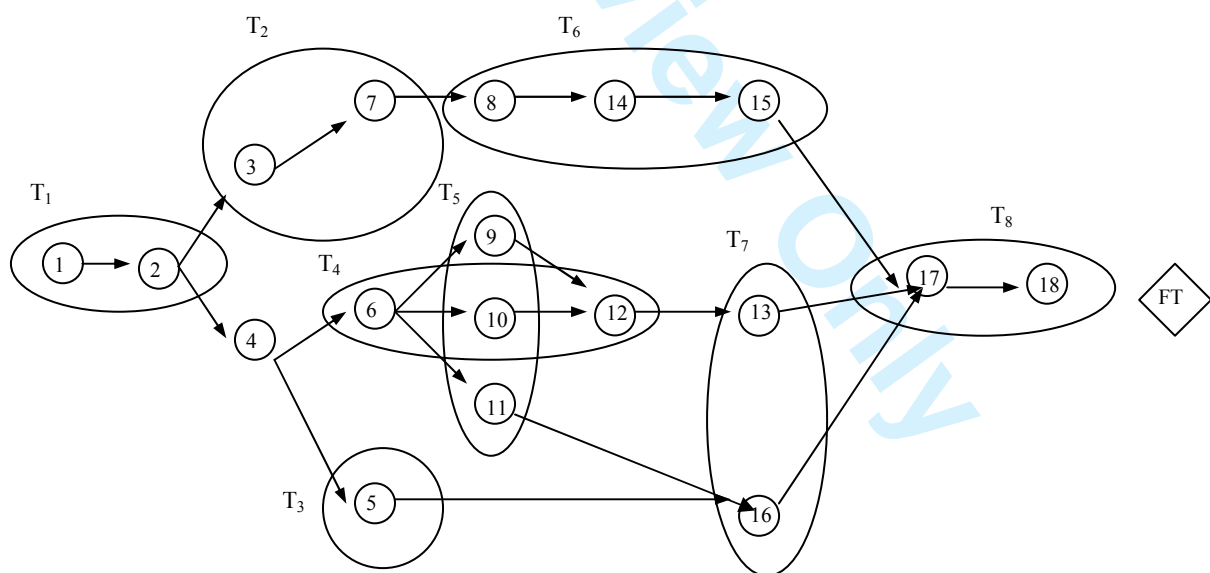


Figure 1: An assembly graph with a set of possible inspections

4.1 Characteristics of the model

Here the characteristics of the model proposed are presented

- Assembly task time is known deterministically and is not affected by the allocation sequence (the tasks and tests already assigned and their relative position)
- Test time is known deterministically and is not affected by the allocation sequence
- Defect rate of each assembly task is known, constant and is not affected by the allocation sequence
- Tests are perfect: no first type or second type inspection error
- Each task can be controlled just by one test in line
- The final test is not in line, and therefore its processing time is not relevant
- The final test checks all tasks
- The reparation cost increases with the distance of the test from the tested task

The first hypothesis, i.e. the fact that assembly tasks time do not depend on the sequence of operations is not always true in reality. Nevertheless it is possible to take into account this by restraining the task sequence. The same can be said for the second hypothesis. The assembly task defect rate has been considered deterministic as it does not impact on the total output and is not the central issue of the problem.

The same holds for the fourth hypothesis.

The hypothesis that each task can be controlled by a test in line only, has been introduced to reduce computational difficulties.

The model considers the fact that reparation cost increases with the distance of the test from the tasks it checks. This is modeled in the position added cost that has been explained above.

4.2 Nomenclature

Here follows a definition of the symbols that have been used

Indexes

n	number of nodes of the assembly diagram, it is the same as the number of tasks to assemble the product
i	task index ($i \in [1, n]$)
K	number of available inspection tests (except the final test); each control can check more than one defect
k	inspection test index ($k \in [1, K]$)
n_{\min}	theoretical minimum number of stations (see appendix)
MaxW	maximum number of stations (see appendix)
j	station index ($j \in [1, \text{MaxW}]$)

Data

CE_i	external failure cost associated to defect i
CF_i	reparation/replacement cost if task i is controlled at the final test
CFT	final test cost
CI	station installation cost (portion of the station installation cost allotted to a unit of output)
$Clin_i$	reparation/replacement cost for in line inspections associated to test i (see appendix)
$Clin_k$	reparation/replacement cost for in line inspections associated to test k (see appendix)
CO_k	cost of test k

$CP_{i,k}$	position cost, it is the added reparation cost that has to be afforded if task i has already been performed in a previous station when test k is done and the product is found to be defective by test k
CT	assembly line cycle time
d_i	defect rate of assembly task i (see appendix)
G_k	probability that the tasks controlled by test k were performed correctly (see appendix)
H	binary $[K,n]$ matrix of non feasibility of the test
$H_{k,i}$	component of H : $H_{k,i}=1$ if the execution of operation i makes test k no more feasible, 0 otherwise
I_k	set of operations controlled by test k (e.g. $I_3 = \{ 1, 4, 6 \}$, test 3 controls the tasks 1, 4 and 6)
M	a very large number
OT	binary $[n,K]$ matrix indicating which tasks are controlled by the tests
$OT_{i,k}$	component of OT matrix; $OT_{i,k}=1$ if task i is controlled by test k, 0 otherwise
PR	$[n,K]$ binary matrix of the test-assembly task position
$PR_{i,k}$	component of the PR matrix: $PR_{i,k}=1$ if having performed operation i before test k is done, determines an increase in reparation cost if the product is found to be defective by test k.
$PT(i)$	set of immediate predecessors of task i
$PI(k)$	set of all the predecessors (tasks) of test k
t_i	time required to perform task i
τ_K	time required to perform test k

Variables

FT	binary variable that equals 1 if the final test is allocated, 0 otherwise
L_i	auxiliary binary variable needed to make the product $FT \cdot TE_i$ linear ($=FT \cdot TE_i$)
NOP_i	number of the station to which task i is assigned
$NTEST_k$	number of the station to which test k is assigned
Q_k	index of allocation of test k: it equals 1 if test k has been allocated to a station, 0 otherwise
$T_{k,j}$	binary variable that equals 1 if test k is assigned to station j ($k \in [1, K]$, $j \in [1, MaxW]$), 0 otherwise
TE_i	index of assembly task inspection: $TE_i=1$ if task i is controlled, (i.e. if one of the tests available to control it is assigned), 0 otherwise
W_j	binary variable that indicates the opening of station j, it equals 1 if at least one operation (task or test) is allocated to the station, 0 otherwise
$Z_{i,j}$	binary variable that equals 1 if task i is assigned to station j ($i \in [1, n]$, $j \in [1, MaxW]$), 0 otherwise
$\Psi_{i,k}$	binary variable which equals 1 if a position added cost has to be calculated as concerns test k and operation i, 0 otherwise

4.3 Objective function

$$\text{Min: SIC} + \text{InC} + \text{EFC} + \text{LRC} + \text{FRC} + \text{TPAC} \quad (1)$$

where

$$\text{SIC} = \text{Station Installation Cost} = CI \cdot \sum_{j=1}^{MaxW} W_j \quad (2)$$

$$\text{InC} = \text{Inspection Cost} = \sum_{k=1}^K CO_k \cdot Q_k + CFT \cdot FT \quad (3)$$

EFC= External Failure Cost= $\sum_{i=1}^n CE_i \cdot (d_i - d_i \cdot TE_i - d_i \cdot FT + d_i \cdot L_i)$ (4)

LRC= In line Reparation/Replacement Cost= $\sum_{k=1}^K Clin_k \cdot Q_k \cdot (1 - G_k)$ (5)

FRC= Final Test Reparation Cost = $\sum_{i=1}^n CF_i \cdot (d_i \cdot FT - d_i \cdot L_i)$ (6)

TPAC= Test Position Added Cost= $\sum_{i=1}^n \sum_{k=1}^K CP_{i,k} \cdot \Psi_{i,k} \cdot (1 - G_k)$ (7)

The value of the objective function represents the unit cost associated to the manufacturing of one product. This cost takes into account relevant costs for our model, that is the Station Installation Cost, the Inspection Cost, the External Failure Cost, the In line Reparation/Replacement Cost, the Final Test Reparation Cost and the Test Position Added Cost. Other costs that would not be modified by the application of our model (as the cost for materials) are not included.

The station installation cost is given by the number of stations times the unit station installation cost (CI) which accounts for soil occupation cost, heating, personnel (if present) and all the other fixed costs. CI has been calculated as the portion of the station installation cost allotted to a unit of output.

The inspection cost is made up of two terms. The first one is the cost to perform a test due to the need of tools and fixtures. The second one represents the cost of the final test, in case it is performed. The cost of time consumption due to performing the test is already considered in the station installation cost, because it affects the number of stations needed .

The external failure cost represents the cost incurred when defective units reach the market. In (4) it is simply the external failure cost times the outgoing defectiveness. It is necessary here to give some explanations about the variable L_i . It is an auxiliary

variable that represents the product of TE_i and FT . As it will be explained afterwards, constraints (18) and (19) force L_i to 1 just when both TE_i and FT equal 1. In table 1 the outgoing defectiveness is indicated as a function of the value of TE_i , FT and L_i .

Table 1: Outgoing defectiveness as a function of TE_i , FT and L_i

TE_i	FT	L_i	Outgoing defectiveness $(d_i - d_i \cdot TE_i - d_i \cdot FT + d_i \cdot L_i)$
0	0	0	d_i
1	0	0	0
0	1	0	0
1	1	1	0

The outgoing defectiveness associated to operation i , equals d_i if task i is controlled neither in line nor at the final test, otherwise it is zero.

The in line reparation/substitution cost has been evaluated as the units entering the reparation station times the reparation/substitution cost. The way the variable G_k can be inferred from the defect rates of the single assembly task (d_i) is explained in the appendix.

Equation (6) calculates the reparation cost after the final test as the product of the detected defectiveness at the final test (associated to all the tasks except the ones already tested in-line) and the reparation cost after the final test. In table 2 the detected defectiveness at the final test has been reported as a function of TE_i , FT and L_i .

Table 2: Detected defectiveness at the final test as a function of TE_i , FT and L_i

TE_i	FT	L_i	Detected Defectiveness at the final test $(d_i \cdot FT - d_i \cdot L_i)$
0	0	0	0
1	0	0	0
0	1	0	d_i
1	1	1	0

As said before the final test is an overall inspection that can find all the defects of a product. The detected defectiveness at the final test is used to evaluate the reparation cost after the final test. If the final test is not performed ($FT=0$), since defective products would not be found, the detected defectiveness at the final test equals 0. The same happens if the task has already been controlled in line ($TE_i=1$): in this case the defectiveness is 0 whether the final test is performed or not. The detected defectiveness at the final test equals d_i if task i is controlled just at the final test.

Equation (7) represents the test position added cost. As said before this cost keeps into account the fact that if a task is not controlled straight after it is done and some other tasks have already been done between the moment in which the task is done and the moment in which it is checked, some extra reparation cost might arise if the product is found to be defective by the test. This is due to the fact that reparation may require partial disassembly and then re-assembly of tasks performed in stations downstream of task inspected but before the test is performed. The test position added cost is computed as the product of the defective units entering a reparation station times the position cost ($CP_{i,k}$). Constraints as in equation (20) forces $\Psi_{i,k}$ to become 1 whenever having performed operation i before test k implies extra-reparation costs if the product is found

to be defective by test k . Through the matrix CP the model considers all the tasks that are performed before test k is done and the different impact that they can have on reparation cost.

4.4 Constraints

The constraints of the model are the following:

$$\sum_{j=1}^{MaxW} Z_{i,j} = 1 \quad i = 1, \dots, n \quad (8)$$

$$Z_{i,j} \leq \sum_{h=1}^j Z_{v,h} \quad i = 1, \dots, n; \quad j = 1, \dots, MaxW; \quad \forall v \in PT(i) \quad (9)$$

$$T_{k,j} \leq \sum_{h=1}^j Z_{v,h} \quad k = 1, \dots, K; \quad j = 1, \dots, MaxW; \quad \forall v \in PI(k) \quad (10)$$

$$T_{k,j} \leq 1 - \sum_{h=1}^j \sum_{i=1}^n Z_{i,h} \cdot H_{k,i} \quad k = 1, \dots, K; \quad j = 1, \dots, MaxW \quad (11)$$

$$\sum_{i=1}^n t_i \cdot Z_{i,j} + \sum_{k=1}^K \tau_k \cdot T_{k,j} \leq CT \cdot W_j \quad j = 1, \dots, MaxW \quad (12)$$

$$W_{j+1} \leq W_j \quad j = 1, \dots, MaxW-1 \quad (13)$$

$$Q_k = \sum_{j=1}^{MaxW} T_{k,j} \quad k = 1, \dots, K \quad (14)$$

$$NTEST_k = \sum_{j=1}^{MaxW} j \cdot T_{k,j} \quad k = 1, \dots, K \quad (15)$$

$$NOP_i = \sum_{j=1}^{MaxW} j \cdot Z_{i,j} \quad i = 1, \dots, n \quad (16)$$

$$TE_i = \sum_{k=1}^K OT_{i,k} \cdot Q_k \quad i = 1, \dots, n \quad (17)$$

$$TE_i + FT - L_i \leq 1 \quad i = 1, \dots, n \quad (18)$$

$$- TE_i - FT + 2 \cdot L_i \leq 0 \quad i = 1, \dots, n \quad (19)$$

$$M \cdot \Psi_{i,k} \geq PR_{i,k} \cdot (NTEST_k - NOP_i) \quad k = 1, \dots, K; \quad i = 1, \dots, n \quad (20)$$

Each task has a constraint of the form of equation (8). These constraints ensure that each task is assigned to exactly one workstation.

Restrictions as in equation (9) forces adherence to precedence restrictions: to allocate an assembly task to a station, it is necessary that all the tasks that are immediate predecessors of this task were previously assigned.

The next restriction expresses the same concept in test precedence relationships.

Exclusion relationships for the tests are satisfied by equation (11): if a task that makes the test unfeasible is allocated, the variable $T_{k,j}$ is forced to zero implying that the test cannot be assigned.

Constraints as in equation (12) ensure that the sum of task time and test time assigned to each workstation does not exceed the cycle time. Besides, equations as in (12) force W_j to 1 if at least a task or a test is allocated to station j .

To avoid that a station is open unless the previous station has already been opened, restrictions as in (13) are set.

Equations as in (14) and (17) define Q_k and TE_i . As TE_i is a binary variable, constraints as in equation (17) impose that every task can be controlled just by one test (in other words it is possible to allocate just one test that controls operation i).

Equations as in (15) and (16) are used to evaluate the test position added cost together with constraints as (20). $NTEST_k$ represents the number of the station to which test k is assigned (0 if test k is not assigned) and NOP_i represents the number of the station to which test i is assigned. $NTEST_k - NOP_i$ is either 0 (if test k and operation i are assigned to the same station), ≥ 1 (if test k is assigned in a station after i) or ≤ 1 (if test k is not assigned or is assigned to a station before the one in which i is assigned). In equations as (20), $\Psi_{i,k}$ is forced to 1 when test k is done after task i has already been assigned in a previous station and PR_{ik} equals 1 (i.e. having performed operation i before test k is performed is a cause of extra reparation cost if the product is found to be defective by

test k). When $\Psi_{i,k}$ is forced to 1, test position added cost is computed in the objective function (see equation 7). It is worth noting that equation (7) considers all tasks for each test thus allowing to compute the contribution to TPAC of all tasks already assigned to stations between a task and its test. This allows to consider an increase in reparation cost as the test is done downstream of the tested task.

Each task has constraints as (18) and (19) necessary to make the product of the two binary variables TE_i and FT linear. They force the binary variable L_i to equal 1 just when TE_i and FT are both equal to 1 and to be zero otherwise.

Further constraints could be added to the model to take into account other requirements connected both to assembly tasks and inspection tasks. For instance on one hand it could be necessary to allocate two assembly tasks to the same station (for example because they require the same expensive equipment); on the other, some assembly operations could be incompatible and thus have to be allocated to different stations (for example an arc welding task and a task requiring the use of an inflammable solvent). Similar considerations could be made for inspection tasks. In all these cases specific constraints can be added to the model.

5. EXPERIMENT DESIGN

A set of simulation runs has been performed to evaluate the advantages of the integrated approach to assembly line balancing and quality management proposed in the model (named as LBQ) in comparison with a sequential approach achievable adapting the models present in literature.

5.1 Benchmark selection

In order to evaluate the real benefits of the model presented, it has been compared with models already existing in literature. In literature there are no models that concurrently address the problem of assembly line balancing and inspection policy definition,

therefore the benchmarks used to evaluate the model derives from the sequential combination of two different algorithms derived from the literature. The first algorithm defines the inspection policy (which tasks to control); the second algorithm takes as input the output of the first algorithm and balances the line by allocating tasks and tests to workstations (see also figure 2).

Based on the above explained principle, two different benchmark models have been designed, they have been named CONF-P and CONF-KL.

CONF-P model first defines the inspection policy by using the same approach proposed by Brytney (1972) or Yum and Mc Dowell (1981, 1987). Indeed these models, among the ones proposed in literature for inspection allocation in multistage production systems, are the ones that suite better the problem proposed. Nevertheless a slight modification of these models has been necessary. Indeed in the above mentioned model, an inspection is available after every production step and it checks all the tasks that has been done until that stage, while in the problem proposed there is a set of possible inspections that control one task or a set of tasks. The costs that are considered while defining the inspection policy are inspection cost, reparation/replacement cost, and external failure cost. The test position added cost is neglected at this stage as it is assumed that, if a test is done, it is performed straight after the task it checks. Using as input the output of the first model, Askin and Standridge's model is used to balance the line thus computing the line installation cost. At the end, the test position added cost is calculated as a result of the position assigned to the tests by the second algorithm.

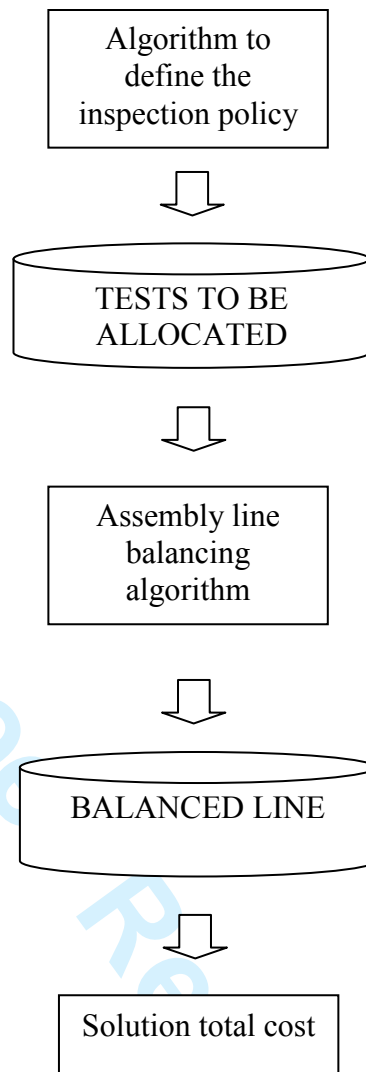


Figure 2: Logical schema of the benchmark models

CONF-KL has the same approach described for CONF-P except for the fact that the costs used in the first algorithm are slightly different. Using the same idea at the basis of Kottas and Lau algorithm (1973) the costs of the tests has been defined as in equations (21).

$$CTEST_k^* = CTEST_k + \frac{\tau_k}{CT} * CI \quad (21)$$

where

k=test index

CTEST_k=test cost as used in CONF-P model

τ_k = time required to perform test k

CT= cycle time

CI= unit station installation cost

Thus the cost of the tests used in the CONF-KL model equals the cost of the tests used in the CONF-P model plus a percentage of the station installation cost based on the rate of time taken to perform the test and the cycle time. CONF-KL should therefore account for the time to perform the test and thus favor, *other things being equal*, short time tests which have a less relevant impact on line balancing.

5.2 Experimental factors

The models have been compared in different situations in order to test their characteristics. In particular the following factors have been identified as the most relevant: the graph type, the station installation cost and the test position cost and have been varied as indicated in table 3.

Table 3: Experimental factors

Parameter	Levels #	Levels
Graph type (GT)	3	Graph A Graph B Graph C
Station installation cost (CI)	3	Low (0,3) Medium (1) High (4)
Position cost ($CP_{i,k}$)	3	Low (0,1) Medium (1) High (4)

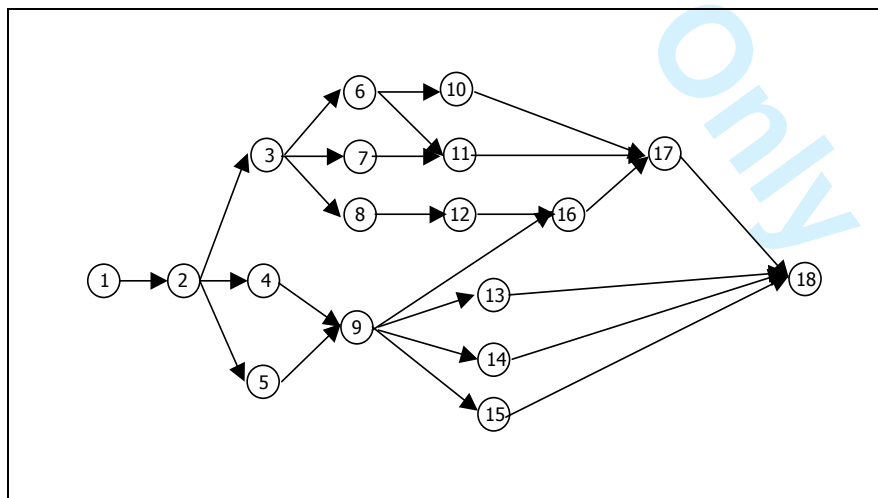


Figure 3: Precedence Graph type A

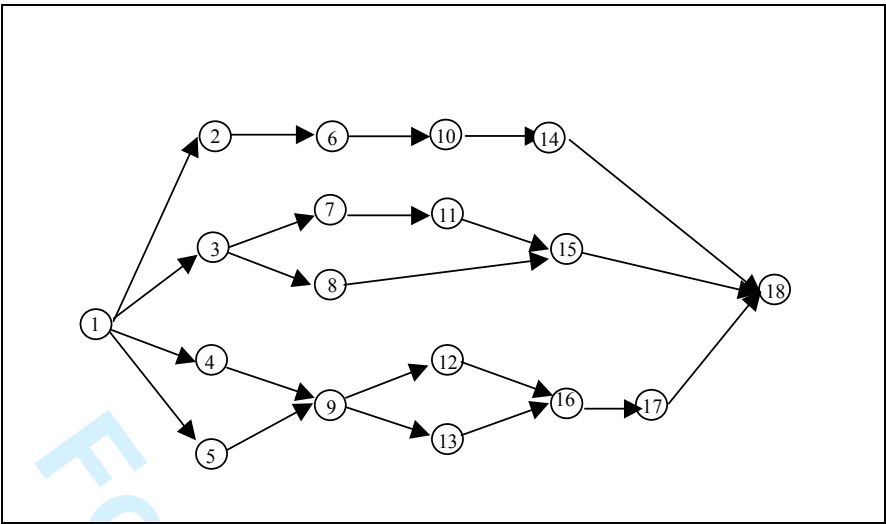


Figure 4: Precedence Graph type B

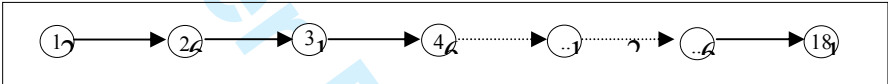


Figure 5: Precedence Graph type C

Table 4: Details about tests

TEST	1	2	3	4	5	6	7	8	9	10	11	12	13	14
TIME	72	56	47	34	71	79	38	65	66	53	30	49	65	45
TASKS CHECKED	2,3,5	7,9,11	2,3	8,16	13,14,15,17	2,3,4,5	4,6	10,11,12,15	6,12,13	6,7,8	15,17	10,11	5,7,9	15,16,17

Graph A and graph B are described in figure 3 and 4 respectively. They represent different precedence relationships but have the same number of assembly operations and the same overall work content. (i.e. the sum of the duration of the 18 assembly tasks is the same for all the graphs). Finally figure 5 represents graph C which is an extreme situation with a serial sequence (the sequence of tasks is fixed).

All the graphs have 18 tasks and 14 possible inspections. The number of tasks and tests

influences the complexity of the problem. The number of tasks and tests used for the simulation are consistent with the models in literature concerning allocation of tests in multistage production systems; for instance Kaspi and Raz (1994) considered a diagram with 8 tasks and 5 tests and in Yum and McDowell (1981) the number of tasks is 10 as well as the number of tests.

As to the station installation cost, it is a relevant parameter because it is connected with the investment necessary to install the line. High value of the station installation cost are typical of a product that requires high expenses for handling equipment during the assembly procedure, or in a situation in which the personnel, the soil occupation cost etc. are very expensive.

A high value for the position cost is used to represent a situation where not finding the defect right after having performed a task, is very costly due to the cost of disassembly and the presence of unrecoverable components etc.

In order to simulate low, medium and high value of the installation cost and the position cost the following procedure has been followed. For each one of those parameters a value has been taken as baseline representing the medium value. High value and low values have been calculated by multiplying the medium value for the parameters in Table 3. Thus, for example, low value and high value of the station installation cost are respectively 30% and 400% of the medium value.

To improve statistical confidence 9 replications have been carried out changing the time needed to perform each task by re-allocating randomly the processing times among the 18 assembly tasks (assembly task times vary from 19 to 159). The overall assembly time is thus maintained constant, so making easier to compare the different performances. Cycle time has been set to 190.

As to the other parameters of the model connected to the tasks, the defect rate of assembly task varies from 0.021 and 0.132, reparation/replacement cost at the final test

range from 138 to 205, the external failure cost associated to a defect varies from 511 to 691. As concerns other costs, the cost of a test varies between 5 and 25, in line repairation cost varies between 126 and 181 while the cost of the final test is 761. The installation cost is 523.

5.3 Performance measures

The different models tested have been compared on the basis of the total cost comprising station installation cost, inspection cost, external failure cost, in line repairation/replacement cost, repairation cost after the final test and test position added cost as they are defined in equation (1).

6. EXPERIMENT RESULT

In figure 6 the overall average total cost (considering all simulation runs) has been reported.

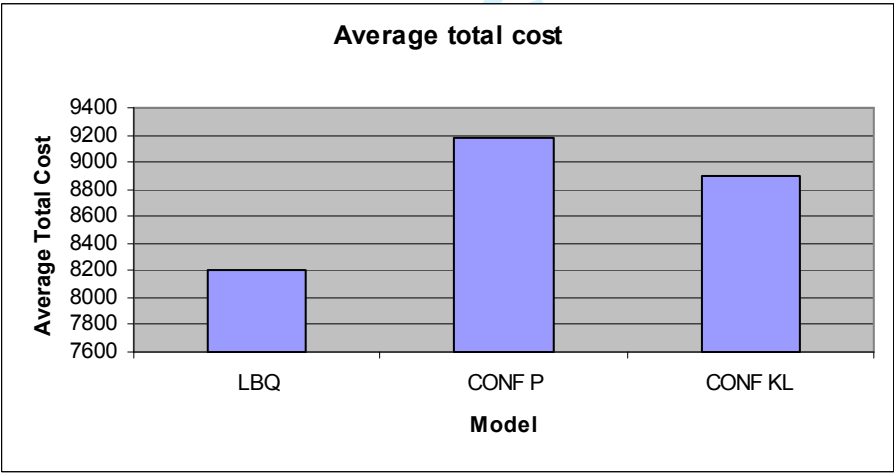


Figure 6: Average total cost

The proposed integrated model LBQ yielded the lowest cost, whilst CONF-P had the worst performance.

CONF-P has been taken as benchmark to evaluate the relative performance of the other two models.

A statistical test on total costs shows that both LBQ and CONF-KL are significantly better than CONF-P at $\alpha=0.05$ (and thus the finding has 95% chance of being true).

The average percentage improvement is 8.7 % for LBQ and 1.3% for CONF-KL.

The average percentage improvement has been calculated in the following way:

$$\%improv = \frac{\sum_{i=1}^{total_number_runs} \frac{Cost(Conf - P) - Cost(LBQ)}{Cost(CONF - P)}}{total_number_runs} \quad (22)$$

The gap for LBQ is quite relevant if we consider that this percentages has been calculated considering the solution total cost. Indeed the number of stations in a line cannot be inferior to the minimum number calculated dividing the sum of all the assembly task processing times by the cycle time. This cost component, which is not negligible, cannot be ascribed to the model. Comparing the improvement achieved to the avoidable costs only, the improvement is much bigger. Anyway, being the total cost easier to understand as an objective value, the performance will be considered always in term of total cost.

It is interesting to highlight that LBQ has performed better than CONF-P in all the simulations and the percentage improvement has varied from a minimum value of 1.6% to a maximum value of 13%.

In order to understand better the influence of the experimental factors, the disaggregated data have been reported hereafter.

6.1 Graph type

In figure 7 the average percentage improvement has been reported as a function of the graph type.

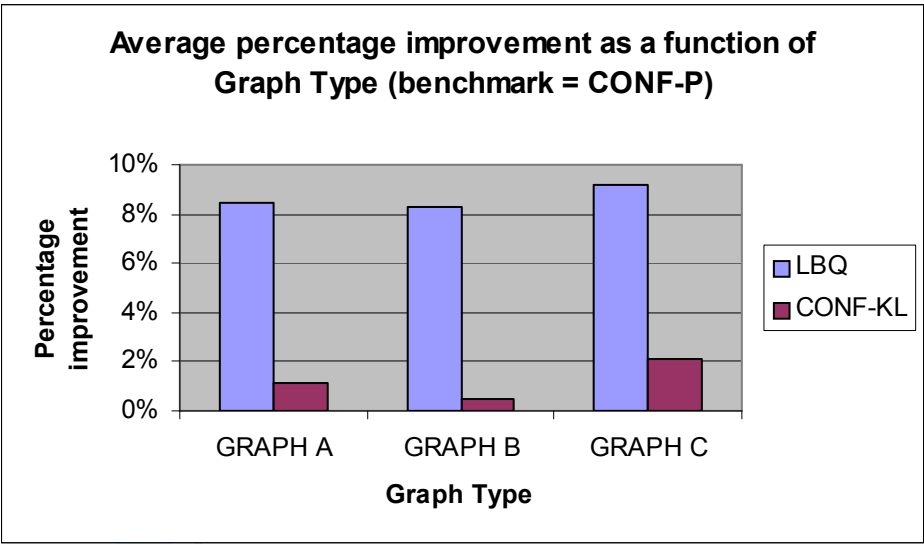


Figure 7: Average improvement in costs as a function of graph type

In all the cases LBQ and CONF-KL performed better than CONF-P and the improvements are statistically significant at $\alpha=0.05$.

The test on the percentage differences calculated on the graph type shows that these differences are not statistically significant. This is true both for LBQ and CONF-KL. LBQ and CONF-KL are better than CONF-P but this does not depend upon the graph type.

6.2 Station installation cost

In figure 8 the percentage improvement has been reported as a function of the station installation cost.

The percentage improvement of LBQ on CONF-P varies from 5.1% for low level of the installation cost to 11.4% for high level of this factor maintaining a value of 9.4% for medium level.

Statistical tests show that LBQ is better than CONF-P and the gap is enhanced when the installation cost raises. CONF-P bad performance is due to the fact that it does not solve the problem globally; LBQ indeed can in some situations allocate the tests so to guarantee an inferior number of stations and thus, the more the installation cost

increases, the bigger savings are possible.

The same can be inferred about CONF-KL that improves its performance as the installation cost increases. This can be explained considering the way tests are appraised in CONF-KL, indeed CONF-KL somehow anticipates the effect of inspection time by attributing to the tests an extra-cost proportional to the processing time and to the installation cost itself.

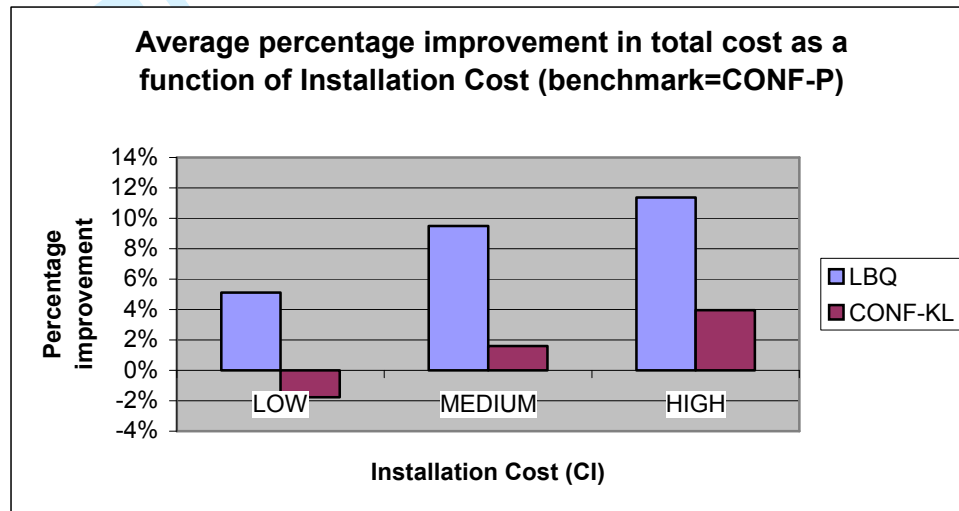


Figure 8: Average improvement in costs as a function of installation cost

6.3 Position cost

In figure 9 the percentage average improvement has been reported as a function of the position cost.

The savings made possible by LBQ range from 7.8% to 9.8%.

The difference between high value of the position cost and medium value is statistically significant whilst the difference between low and medium level is not. We can conclude that LBQ advantage increases as the position cost becomes sufficiently higher.

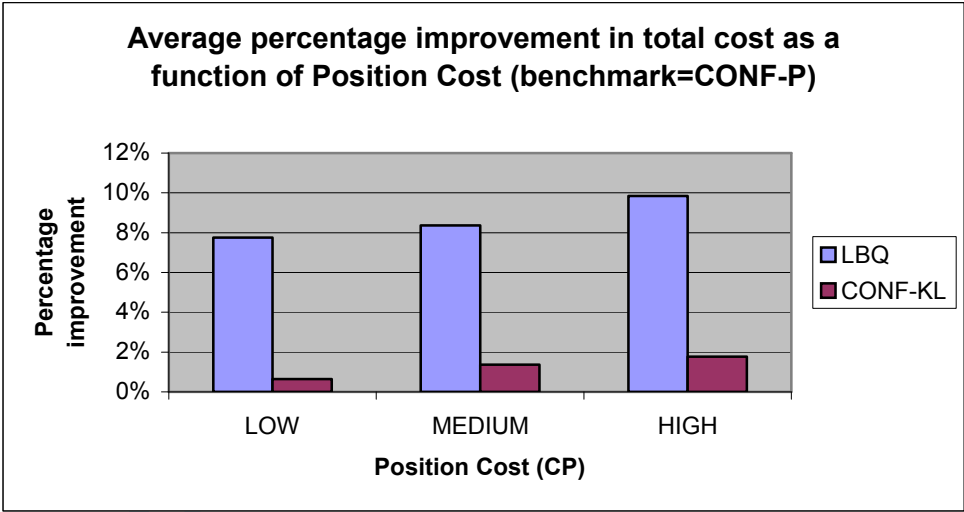


Figure 9: Average improvement as a function of test position cost

The advantage of LBQ lies in the optimisation of the trade off between good line balancing and the costs due to the allocation of tests not straight after the assembly tasks they check. This is not possible for the other two models that address line balancing and inspection policy in a sequential way.

Also considering the test position cost, CONF-KL performs slightly better than CONF-P. The percentage difference is significant only between low value of the station installation cost and high value. A possible explanation of the better performance of CONF-KL compared to CONF-P can be the fact that CONF-KL, all the rest being equal, favours short time tests. It is then more likely that these tests can be allocated to the same station of the tasks they control.

7. CONCLUSIONS

In literature, line balancing and quality control policies are addressed as two different problems, nevertheless the definition of the inspection policy has a very deep impact on line balancing. Thus an integrated approach to line balancing and inspection policy definition can give relevant benefits. In this paper a model for integrating line balancing and quality control policy selection has been proposed and tested.

The model permits to balance an assembly line concurrently allocating assembly tasks and inspection tasks so to minimise the total cost of quality and installation. The model takes into account the cycle time, inspection tasks times, tasks defect rate, precedence and exclusion relationships for tests. Objective of the model is to minimize an objective function that considers inspection costs, reparation/substitution cost for in line controls and the final test, external failure cost, position added cost (to account for extra reparation costs connected to the relative position of a test and the tasks it checks) and installation cost (connected with the total number of workstations that is necessary to set up).

The model has been tested by comparing it with two benchmark models that address the problem of inspection policy definition and assembly line balancing in a sequential way so that first the inspection policy is defined and then tests and tasks are allocated to different workstations. The result of the simulation has showed that the method proposed always achieves a better performance than the benchmark models. Statistical analysis shows that the savings made possible by the model proposed increase as the position cost and the installation cost raise. In this situation the model is especially suitable. Nevertheless also for low values of this factor, savings are significant.

The model could be developed so to adapt it to multi-model lines, consider stochastic times for assembly tasks and inspections. Moreover heuristics could be developed to solve quickly large problems integrating inspection policy design into assembly line balancing.

The model proposed considers several factors: Station Installation Cost, Inspection Cost, External Failure Cost, In line Reparation/Replacement Cost, Final Test Reparation Cost, Test Position Added Cost. All these factors reflect different objectives and they have been weighted estimating their economical impact (e.g. the impact of selling a

defective item), so to reach an objective function which is the sum of a number of different costs. Other types of weights could be adopted.

On a broader basis the model could also be developed to consider the possibility to evaluate alternative graphs and/or task duration, for the same product, so to consider alternative product design options and/or alternative process/technology solutions; this approach indeed is also very promising (Portioli-Staudacher, Singh, 1996).

APPENDIX

Estimation of n_{min} and $MaxW$

$$n_{min} = \left\lceil \frac{\sum_{i=1,n} t_i}{CT} \right\rceil \tag{23}$$

$$MaxW = n + k \tag{24}$$

n_{min} is used only for computational purposes and can be estimated assuming to allocate only assembly tasks.

$MaxW$ is estimated by assuming to allocate, besides all the tasks, all the tests and considering the worst condition for balancing purposes, that is supposing that it is necessary to open a new station for every operation.

Relationships between $Clin_i$ and $Clin_k$

$Clin_k$ (in-line reparation cost associated to test k) can be calculated as follows:

$$Clin_k = FC_k + \sum_{i \in I_k} Clin_i * d_i \tag{25}$$

FC_k =fixed cost associated to test k

Estimation of d_i and relationships between d_i and G_k

We will analyze here in details how we could estimate the defect rate of an assembly task, one of the primary parameters of the model.

Let us call $A_{j,i}$ the j cause of defectiveness for the operation i and CS_i the total number of defectiveness causes. Assuming that the causes of defectiveness are independent, the probability d_i could be written as:

$$d_i = P(\text{operation } i \text{ leads to a defect}) = 1 - P(\text{no defect causes happen}) = 1 - \prod_{j=1}^{CS_i} (1 - p_{A_{j,i}}) \quad (26)$$

If we have an estimation of the probability associated to all the causes of defectiveness, it is then possible to assess the defect rate of every assembly task.

As a result it is possible to estimate the probability G_k that gives the probability that the assembly tasks controlled by test k were been performed properly. Using the hypothesis of independence of the defect rate from the allocation sequence, G_k is expressed by the following formula:

$$G_k = \prod_{\forall i \in I_k} (1 - d_i) \quad (27)$$

REFERENCES

Askin, R.G., Standridge, C.R., 1993, Modeling and analysis of manufacturing systems, Wiley&Sons, New York.

Ballou, D., Pazer, H.L., 1982, The impact of inspector fallibility on the inspection policy in serial production systems, *Management Science*, 28(4), 387-399.

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

Britney, R., 1972, Optimal screening plans for non serial production systems, *Management Science*, 18(9), 550-559.

Eppen, G.D., Hurst, G., 1974, Optimal location of inspection stations in a multistage production process, *Management Science*, 20(8), 1194-1200.

Garcia-Diaz, A., Foster, J.W., Bonyuet, M., 1984, Dynamic programming analysis of special multi-stage inspection systems, 1984, *IEE Transactions*, 16, 115-125.

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*, 32(1), 209-221.

Kaspi, M., Raz, T., 1994, Optimal sequencing of production and inspection operations, *International Journal of Production Research*, 32(1), 209-221

1
2
3
4 Kottas, J.,F., Lau, H.,S., 1973, A cost oriented approach to stochastic line balancing,
5
6 A.I.I.E. Trans. 5,164.
7
8
9

10
11 Lee, J., Unnikrishnan, S., 1998, Planning quality inspection operations in multistage
12 manufacturing system with inspection errors, International Journal of Production
13 Research, 36, 141-155.
14
15
16
17

18
19
20
21 Lindsay, G.,F., Bishop, A.,B., 1964, Allocation of screening inspection effort – a
22 dynamic programming approach, [Management Science](#), 10, 342.
23
24
25
26

27
28 Lubicz, M., 1983, On the problem of optimization of a quality inspection process
29 structure, International Journal of Production Research, 21(3), 369-381.
30
31
32
33

34
35 Nof, S.Y., Wilhelm, W.E., Warnecke, H.-J., 1997, Industrial Assembly, Chapman and
36 Hall.
37
38
39

40
41
42 Portioli-Staudacher, A., Singh, N., 1997, A framework for integration of production
43 management issues and concurrent engineering, International Journal of Systems
44 Science, Vol. 28, N. 9, 877-888
45
46
47
48

49
50
51 Portioli-Staudacher, 1999, A., A concurrent engineering approach to assembly line
52 balancing, International Conference on industrial engineering and production
53 management, Glasgow UK 12-15/7, 1999
54
55
56
57
58
59
60

Portioli-Staudacher, A., Landeghem, H.,V., Mappelli, M., Redaelli, C., E., 2003, Implementation of concurrent engineering: a survey in Italy and Belgium, Robotics and Computer-Integrated manufacturing, Vol. 19, N. 3, 225-238

Pruzan, P.M., Jackson, J., T.,R., 1967, A dynamic programming application in production line inspection, Technometrics, 9, 73-81.

Rau, H., Chu, Y.,H., 2005, Inspection allocation planning with two types of workstations : WVD and WAD, International Journal of Advanced Manufacturing Technology, 25, 947-953

Raz,T., Kaspi, M., 1991, Location and sequencing of imperfect inspection operations in serial multi-stage production systems, International Journal of Production Research, 29(8), 1645-1659.

Rekiek, B., Dolgui, A., Delchambre, A., Bratcu, A. 2002, Annual reviews in Control, 26, 163-174.

Salveson, M.E., 1955, The assembly line balancing problem, Journal of Industrial Engineering, 6(3), 18-25.

Scholl,A., Balancing and Sequencing of Assembly Lines, 1995, Physica-Verlag, Heidelberg, Germany.

Scholl, A., Becker, C., State-of-the-art exact and heuristic solution procedures for simple assembly line balancing, *European Journal of Operational Research* 168 (2006) 666–693.

Shiau, Y.,R., 2002, Inspection resource allocation in a multistage manufacturing system with an inspection error model, *International Journal of Production Research*, 40(8), 1787-1806.

We-Min Chow, 1990, *Assembly Line Design, Methodology and Applications*, Marcel Dekker, Inc.

Yum, B.,J., McDowell, E.D., 1981, The optimal allocation of inspection effort in a class of nonserial production systems, *AIIE Transactions*, 13(4), 285-293.

Yum, B.,J., McDowell, E.D., 1987, Optimal inspection policies in a serial production system including scrap, rework and repair: An MILP approach, *International Journal of Production Research*, 25(10), 1451-1464