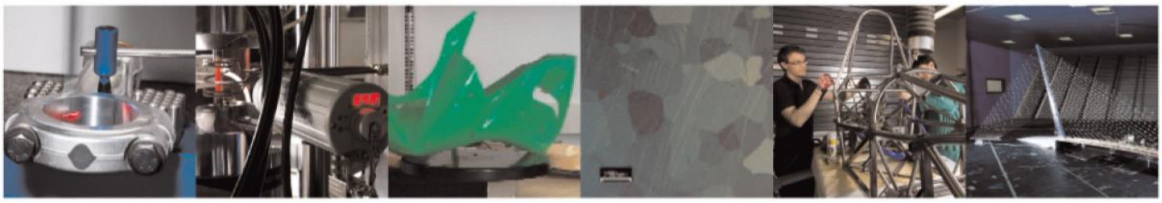
POLITECNICO MILANO 1863

DIPARTIMENTO DI MECCANICA



Robust Improvement Planning of Automated Multi-stage Manufacturing Systems

M.C. Magnanini, T.A.M. Tolio

This is a post-peer-review, pre-copyedit version of an article published in Lecture Notes in Mechanical Engineering. The final authenticated version is available online at: http://dx.doi.org/10.1007/978-3-030-82627-7_4

DIPARTIMENTO DI MECCANICA ■ POLITECNICO DI MILANO
via G. La Masa, 1 ■ 20156 Milano ■ EMAIL (PEC): pecmecc@cert.polimi.it
http://www.mecc.polimi.it
Rev. 0

Robust improvement planning of automated multi-stage manufacturing systems

Maria Chiara Magnanini[1], Tullio A.M. Tolio[1]
[1] Department of Mechanical Engineering, Politecnico di Milano, Milano, Italy
mariachiara.magnanini@polimi.it

Abstract. Automated multi-stage manufacturing systems serve as the backbone of mass industrial production. Increasing the efficiency of these systems represents a key challenge for manufacturing companies, which continuously cope with planning the sequence of optimal improvement actions according to budget and implementation time constraints. Improvement actions related to different areas, as quality, maintenance, logistics, are usually evaluated independently among each other. Recent developments in data gathering support the evaluation of the effect of improvement actions at local level, i.e. single machine, without accounting for the interactions at system-level among machines. This work presents an optimization method for the sequencing of improvement actions in automated multi-stage manufacturing systems. It combines dynamic programming with a stochastic analytical model for the performance evaluation of manufacturing systems. Results from a real industrial case in the furniture sector prove the usefulness of this novel methodology, compared to traditional bottleneck identification and improvement.

Keywords: Performance evaluation, Manufacturing systems, Optimization..

1 Introduction

Automated multi-stage manufacturing systems serve as the backbone of mass industrial production. The objective of production efficiency represents a constant challenge for manufacturing companies. As a consequence, continuous improvement of manufacturing systems characterizes a day-by-day activity, in which entire teams are constantly dedicated to the identification and analysis of improvement strategies and connected implementation activities. Moreover, the context in which manufacturing companies are operating is more and more dynamic. Technological and digital innovations are continuously pushing manufacturing systems to change and adapt to constantly new conditions, in order to remain competitive. Recent advances in technologies, data gathering and control systems have opened new opportunities to evaluate and manage production systems. Indeed, Industrial Internet of Things (IIoT) makes available system data which can be used to feed effective decision support tools.

Strategies for manufacturing system improvement involve decisions at different levels having an impact on different time horizons. Often, companies have to choose among many different improvement actions. These improvement actions are usually focused, i.e. they refer to a single machine or a limited portions of the line while having an impact on the whole system. They require implementation effort and time before becoming effective, therefore, the dynamic performance of the manufacturing system strongly depends on their sequencing and timing. For instance, the replacement of actuators from pneumatic to electric implies machine reconfiguration, similarly, the installation of new sensor technology includes sensory network implementation and software system set up. Therefore, the positive effect of these actions requires time. Moreover actions are not only related to technology but involve also the operators and system organization. For instance, a training course for maintenance operators could reduce the reaction time to a failure and therefore quickly improve the performance but requires learning time.

This work aims at providing a novel optimization method for the robust sequencing of improvement actions in automated multi-stage manufacturing systems (MMS). The scientific relevance of the topic is shown in Section 2, where the related state of the art is analyzed. Then, the problem statement is presented and analyzed in Section 3. The solution methodology is based on the integration of dynamic programming and stochastic performance evaluation model, and it is presented in details in Section 4. Section 5 shows numerical results with respect to existing industrial practices. Conclusion and future research are depicted in Section 6.

2 State of the art

The literature related to planning of improvement actions in automated multi-stage manufacturing systems is quite sparse. Indeed, most of the works focus separately on the following aspects: (i) identification of the bottleneck machine, (ii) selection of improvement strategies. Therefore, the following Sections are devoted to the analysis of each of these aspects.

2.1 Bottleneck identification in multi-stage manufacturing systems

The continuous improvement through bottleneck identification is one of the most common industrial practices. Also, the bottleneck identification is a well-known research problem. The basic idea is to identify the bottleneck machine, improve its performance which implies shifting the bottleneck. The new bottleneck is in turn improved and step by step the performance of the whole line is improved. This methodology can be applied iteratively on a system, and requires the sequential identification of improvement actions. This cyclic procedure can be referred as Bottleneck Release.

For the identification of the bottleneck machine(s), mainly two possible approaches can be followed. The first approach is based on using performance evaluation models (e.g. approximate analytical models or Discrete Event Simulation) to assess the main performance of the line while introducing small changes on the features of the machines. Once the increment in production due these changes are obtained, it is possible to identify the bottleneck empirically by identifying the gradient. This methodology is explained in dept in [1,2], where different types of bottlenecks are classified on the basis of the main limiting factor (the speed of a machine or the reliability of the machine). The other main approach to identify the bottleneck machine is the data driven approach. In this approach, the data used to identify the bottleneck is obtained from the real production line. Using information as blocking times, starvation times, blocking frequency, starvation frequency and inventory levels of the different buffers it is possible to determine the position of the bottleneck [3].

2.2 Selection and planning of improvement strategies

With respect to the selection and planning of improvement strategies, works are mainly divided in quantitative and qualitative approaches. In general, most of the approaches select the order of improvement actions on the basis on the most critical action, i.e. the one that guarantees the highest improvement of the overall performance. For instance, [4] have suggested a sequential approach that is made by eight repetitive steps. the pre-study gather information about the company (data collection). Then, the problem identification needs to be carried out. Next, process mapping should be done to generate an overall picture of the current problem. After that, measurement of critical factors are done. With all these phases, it is possible to start with the selection of the best candidate action to be implemented in the system, and finally check the results of the improvement. Than the whole improvement process is started again.

Most of the works in the literature are devoted to the analysis of one specific production pillar, e.g. quality, maintenance or production operations. Therefore, each proposed method is quite customized on the application, with restrictive assumptions.

For instance, [5] introduce a methodology to rank maintenance initiatives under time and budget limitations to guarantee that the improvement with the highest return on investment is implemented first. This methodology makes use of three indexes to rank activities namely investment, time to completion and budgeting portion of each alternative. Another approach, is proposed by [6], who implemented an hybrid capability maturity model integration - six sigma model to find possible processes improvement actions and select and implement them with a decreasing level of criticality until a desired specific performance reach a defined sigma level of confidence. Another example is the one shown by [7], who developed a mathematical framework to increase the process yield of several stations in the best sequence in order to maximize profit. This succession is understood as a step-wise yield improvement path, were each discontinuity represent an event, thus this approach is connected with a time frame. Under this methodology, the focus is putting on how to improve the yield in each station and thus, how to manage the available budget per unit time among inspection stations. The proposed procedure is characterized by an iterative analysis of the action set, searching for the highest gradient of the expected net profit with respect to the investment.

More in general, related literature is devoted to the identification of indicators for continuous improvement in manufacturing systems [8,9], or to the minimization of ramp-up under predefined target performance [10,11].

3 Problem description

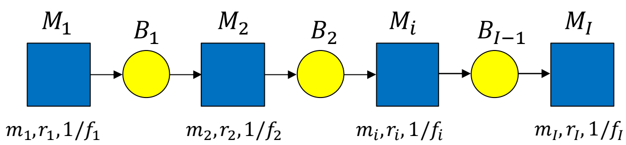


Fig. 1 Graphical representation of a multi-stage manufacturing system.

Notation	
i	production stage (i=1... I)
s	decision time (s=1, … S)
Decision variables	
$u_s \in A$	action selected at time s
$U(s) = \{u_1, \dots, u_S\}$	sequence of actions implemented at time s
Support variables	
$P(s) = [m_i: i = 1, \dots, I]$	vector of production rates at time s
$R(s) = [r_i: i = 1, \dots, I]$	vector of repair rates at time s
$F(s) = [f_i: i = 1, \dots, I]$	vector of mean time to failures at time s
$\mathrm{X}(s) = [P;\ R;\ F]$	vector of system configuration parameters at time s
$Z(s) = \{z_k(1), \dots, z_k(s)\}$	set of combinations of actions not yet implemented at time s
Parameters	
P^0	vector of initial production rates
R^0	vector of initial repair rates
F^0	vector of initial mean time to failures
$A = \{a_1, \dots, a_j\}$	set of improvement actions
$\Delta\mathrm{X}(a_j)$	vector of parameter change due to action a_j
$d(a_j)$	duration of the implementation for the action a_j at time s
th^0	base throughput (e.g. when no action is implemented)
th^*	final throughput (e.g. when all actions are implemented)
Performance measures	
$th_{z_k(s)}$	throughput for not yet implemented combination of actions $z_k(s) \in Z(s)$ at time s
$TH_{U(s)}$	cumulated throughput for sequence of actions $U(s)$ at time s
$\overline{TH}_{U(s)}$	cumulated lost throughput for sequence of actions $U(s)$ at time s

The reference system is described as a Multi-stage Manufacturing System (MMS) with finite buffer capacity. In Figure 1, the manufacturing serial line is shown, where blue squares represent machines and yellow circles represent buffers. Each machine is characterized by production rate *m*, failure rate *1/f* and repair rate *r*. The following list of assumptions holds for the MMS.

MMS: Assumptions

a) The first machine is never starved and the last machine is never blocked;
b) Processing times of the machines are deterministic and may be different among machines;
c) Machines are unreliable and may fail in different modes, with operation-dependent failures;
d) Time to failure and time to repair have a general distribution;
e) Load and unload times are negligible;
f) Parts arrive from outside and leave the system after being processed;
g) The buffer capacity is finite.
h) The system is asynchronous, i.e., each machine can start or finish at any time without synchronization with the other machines.
i) The dispatching policy is Fist In First Out (FIFO);
j) Parts are not scrapped or reworked.

These characteristics are shared by a vast majority of systems that can be found in real manufacturing plants.

It is assumed that for the reference system, a set A of improvement actions a_j has been identified. Each improvement action a_j defines a specific change in a set of parameters of the system and is implemented in a time lag $d(a_j)$. The objective of the problem consists in selecting the sequence of improvement actions maximizing the expected cumulative production quantity. The following list of assumptions holds for the optimization problem.

Optimization problem: Assumptions

a) Improvement actions are independent among each other;
b) Each improvement action of the set must be implemented;
c) Only one improvement action at a time can be implemented.

3.1 Problem statement and discussion

The objective of the problem consists in selecting the sequence of improvement actions maximizing the expected cumulative production quantity. The problem can be visually represented as in Figure 2. Starting from an initial configuration with performance th0, each improvement action has a positive effect on the system throughput, which becomes available only at the end of the duration of the implementation time. Therefore, the resulting throughput function is piecewise linear with respect to the time.

The cumulative production quantity is represented by the area under the curve (red striped area in Figure 2). Indeed, this quantity can be easily obtained by integrating the piecewise linear throughput function.

At each decision time s, a new improvement action a_j from the set is implemented. After the time corresponding to the duration $d(a_j)$, the system throughput is improved. Therefore the throughput at a given time depends on all the actions which have been implemented until that moment. Indeed, different sequences of actions lead to alternative improvement paths.

It makes sense to assume that the set of improvement actions includes both minor and major actions, e.g. actions which are fast to be implemented but they have a minor effect on the configuration parameters of one machine, as well as actions which are not so fast to be implemented but they have a major effect on the configuration parameters of one machine. Moreover, the effect of a single improvement action on the system performance can be evaluated only if all the actions which have been implemented so far are known.

It is interesting to notice that a clear strategy cannot be identified a priori. Indeed, even considering only a finite set of improvement actions to be implemented, there are no general recommendations about whether it is better to implement small actions first, or give precedence to major actions. In this problem, the optimization strategy depends on both aspects: expected throughput improvement and action duration. Also some combinations of actions unlock the full potential improvement which cannot be completely exploited by considering them separately.

Therefore, the optimal improvement strategy should push the system to the implementation of the right action at the right time, i.e. when the improvement action can be more effective with respect to the sequence.

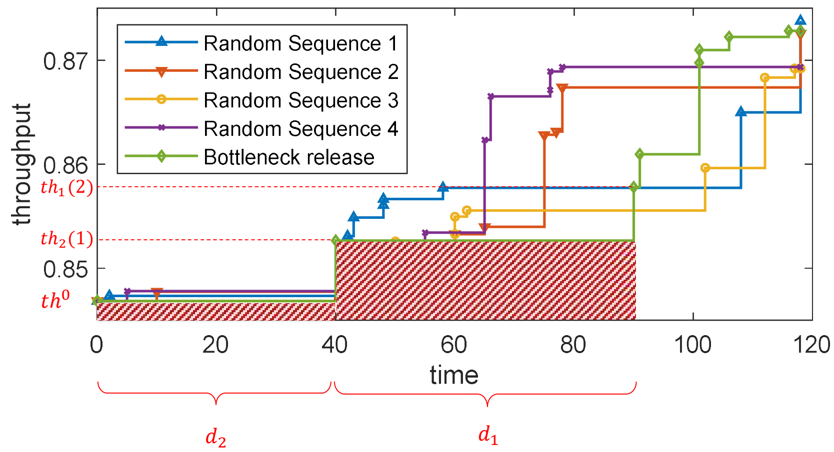


Fig. 2 Throughput as a function of time for alternative improvement sequences and representation of the cumulative production quantity as area under the curve.

The general optimization problem can be framed as a combinatorial problem of finding the optimal permutation of a given set of actions. As well known, this type of problem is NP-hard, and the complexity grows almost exponentially with the number of alternatives. For instance, the example shown in Figure 2 is based on a set of 8 improvement actions, which provides a solution space of 40320 different sequences to be evaluated. Therefore, the main goal of optimization algorithms and heuristics in this problem field is related to the identification of a (sub) optimal solution by evaluating a subset of the solution space. In particular, search methods and metaheuristics are used to identify suitable candidate solutions at each evaluation step s as shown in Figure 3.

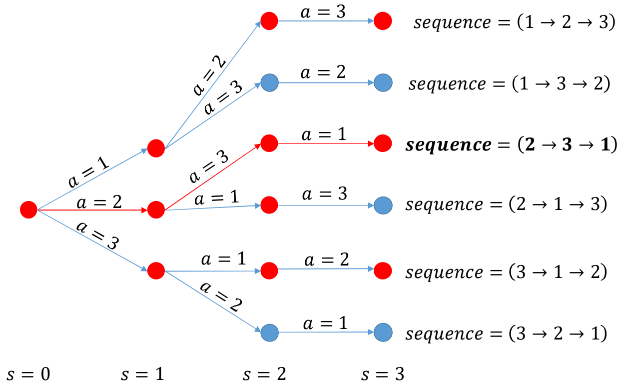


Fig. 3 Overview of solution space for a 3-alternative optimization problem.

4 Methods

4.1 Overview of the approach

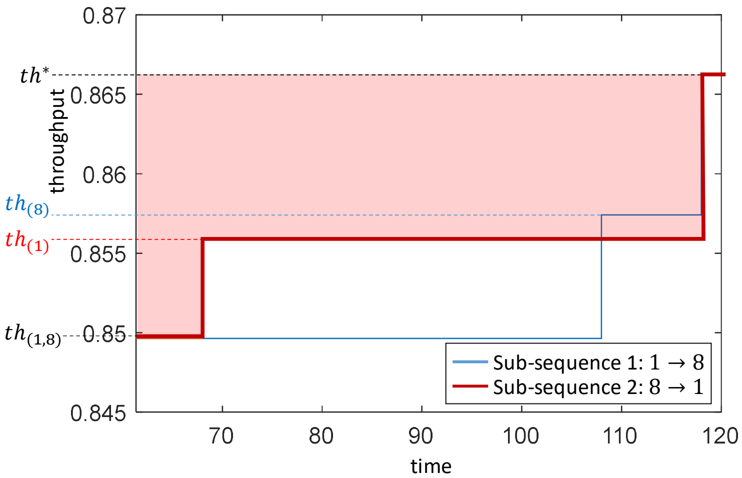


Fig. 4 Representation of the proposed optimization problem.

The problem is approached by means of dynamic programming. Some specific features of the problem make this solution approach viable. Indeed, it can be noticed that the performance in terms of throughput in a given configuration, i.e. for a given sub-set of already implemented improvement actions, does not depend on the specific sequence used to implement the actions. This can be noticed in Figure 4, where the last two steps of two alternative sequence with same set actions is shown. In particular, all actions from 2 to 7 have been implemented, and only actions 1 and 8 are missing. It is shown that the throughput of the first sequence (blue line) when only 1 and 8 are missing is the same as the one of the second sequence (red line) where the same actions have not been implemented yet (indicated as $th(1,8)$ in Figure). Even if both sequence at the end bring the same value of throughput, it can be seen that one permutation dominates the other one. That is, the area above the piecewise linear throughput function is smaller when action 8 is performed first, and then action 1 is performed as last (sequence 1: red line), with respect to that one of sequence 2 (blue line), where action 1 is performed first and action 8 as last one.

Hence, the problem can be transformed in a minimization problem, where the goal is to minimize the area above the piece-wise linear throughput function. This area can be obtained by simply multiplying the duration and the throughput increment of the selected improvement action, with respect to the previous one.

Indeed, the problem can be approached by evaluating the areas due to the couples of improvement actions, and by selecting the minimum one. At each evaluation step, the number of evaluated points is equal to the binomial coefficient. For instance, in a 5-alternative problem, the total number of permutations is 120, but if the dynamic programming procedure is implemented, the total number of evaluation is 24.

4.2 Dynamic programming formulation

Grounding on the explanation provided, the two formulations for the proposed problem are equivalent.

Formulation 1 (maximization). For a finite set of improvement actions $A = \{a_1, \ldots, a_j\}$ acting on the set of configuration parameters $X(s)$ and requiring an implementation time equal to $d(a_j)$, find the sequence $U(S) = \{u_1, u_s, \ldots, u_S, u_s \in A\}$ which maximizes the cumulated throughput $TH_{U(S)}$.

Formulation 2 (minimization). For a finite set of actions $A = \{a_1, \ldots, a_j\}$ acting on the set of configuration parameters $X(s)$ and requiring an implementation time equal to $d(a_j)$, find the sequence $U(S) = \{u_1, u_s, \ldots, u_S, u_s \in A\}$ which minimizes the cumulated lost throughput $\overline{TH}_{U(S)}$.

Therefore, the following dynamic programming approach is used to solve *Formulation 2 (minimization)*.

Step 1. At each decision time s, for all possible combinations $z_k(s) \in Z(s)$, the throughput $th_{z_k(s)}$ is computed by means of the performance evaluation model.

Step 2. For each combination $z_k(s) \in Z(s)$ and for each combination $z_i(s+1) \in Z(s+1)$, so that $z_i(s+1) \subset z_k(s)$, there exists only one arc $\tau_{k,i}$, corresponding to the action $a(\tau_{k,i})$ with duration $d(a(\tau_{k,i}))$. The cost of the arc $\tau_{k,i}$ is computed as

$$c(\tau_{k,i}) = th_{z_k(s)} \cdot d\left(a(\tau_{k,i})\right)$$

Step 3. Starting from decision time S and going backward to the first decision time $s = 1$, the cost-to-go function for each combination $z_k(s)$ at each decision time s is computed as

$$C(z_k(s)) = \min_{i, z_i(s+1) \subset z_k(s)} \left(c(\tau_{k,i}) + C(z_i(s+1))\right)$$

The sequence of the minimum cost-to-go functions on s minimizes the cumulated lost throughput $\overline{TH}(S)$, and according to *Formulation 2* of the problem it returns the optimal sequence of improvement actions $U(S)$.

4.3 Performance evaluation model

The performance measures such as system throughput, but also average level of the buffers and steady-state probabilities can be obtained with any performance evaluation model. In this work, a stochastic analytical model for the performance evaluation of serial lines decoupled by finite-capacity buffers is used as introduced in [12]. The stochastic analytical model is based on continuous-time mixed discrete- and continuous-state Markov Chains. Each machine is therefore described by a state-based representation where process dynamics are captured at machine level. An example of Markov Chain of a machine with one operational state, namely ‘up’, and one failure state, namely ‘down’, is represented in Figure 5.

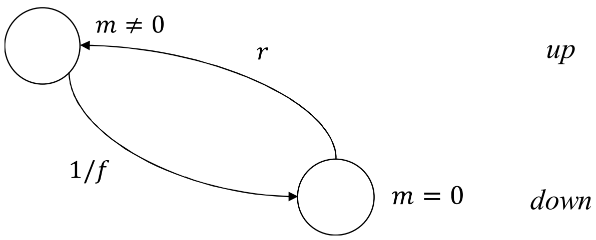


Figure 5. Markov Chain for a single up-single down machine.

Therefore, the whole system is described by the vector of configuration parameters $X = [P; R; F]$, where each vector includes production rates m_i, repair rates r_i and mean times to failure f_i.

Each improvement action a_j is described with its effect on the configuration parameters by the vector $\Delta X(a_j)$. Then, the configuration parameters of the system when the action a_j is implemented, are $X(s) = X(s-1) + \Delta X(a_j)$, where $X(0) = [P^0; R^0; F^0]$. It should be noticed that defining the improvement actions in this way allows to consider the following cases:

1. The improvement action acts on only one parameter of a single machine, e.g. production rate adjustment;
2. The improvement action acts on one whole machine, e.g. replacing it;
3. The improvement action acts on distributed configuration parameters along the line, e.g. replacing line actuators, or improving the sensor network.

In this model, so-called decomposition equations based on the system dynamics model the propagation of effect, i.e. blocking and starvation, along the stages. A linear system of differential equations is solved by a numerical algorithm in order to evaluate the system performance in terms of throughput, average buffer level and steady-state probabilities [12].

4.4 Optimization algorithm

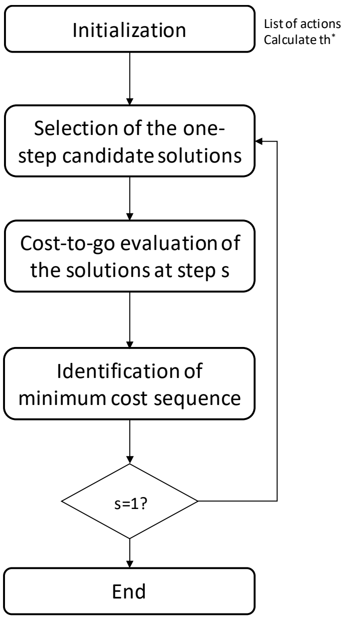


Fig. 6 Graphical representation of the optimization algorithm.

Figure 6 depicts the graphical representation of the optimization algorithm. Indeed, starting from the last step where all actions have been already implemented, iteratively a sub-set of candidate solutions at one step is identified and the configurations evaluated.

Then, the cost-to-go values are computed for each pairs of the candidate solutions in order to identify the sub-sequence of minimum cost. The procedure is reiterated until the initial step is reached.

5 Numerical results

5.1 Comparison with Bottleneck Release

One of the most common used strategies in manufacturing companies for continuous improvement is the Bottleneck Release strategy. In this strategy, at each decision moment, the bottleneck machine is identified and improvement actions are implemented on it until the bottleneck moves to another machine. Therefore, in this Section the proposed method and the Bottleneck Release are compared in terms of optimal solution.

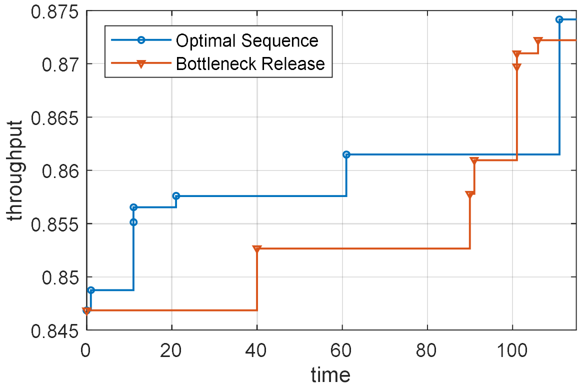


Fig. 7. Comparison in the improvement sequencing between the proposed model and the Bottleneck Release.

Figure 7 shows the results from an example with 8 improvement actions on a MMS. As it can be noticed, the Bottleneck Release does not consider the effect of the improvement actions on the system, nor the consequences of the duration of the implementation. Therefore, it might suggest the implementation of an improvement action on the bottleneck which has only a minor effect, or, as it can be seen in Figure 5, the implementation of an improvement action whose effects can be observed only after some time.

On the other hand, the proposed optimization model accounts for both aspects: effect of improvement actions and duration of the implementation. Also the method considers the positive interactions among improvement actions. Hence, when the proposed model is used, the optimal sequence of improvement actions is found, guaranteeing the maximization of the production quantity over time. The gain in the cumulated production quantity can be more than 3%, with respect to the Bottleneck Release.

6 Industrial case-study

6.1 Description of the MMS

The approach introduced in the paper has been used to evaluate the optimal improvement strategy in a real manufacturing line.

The reference case has been described in [13]. The company is an Italian manufacturer that produces drawers for personalized kitchens. The sides of the drawers are assembled by the production line shown in Figure 8, having a high level of automation. Four stages can be identified in the line. Linear guiderails serve as buffers in the line, having capacity proportional to their length.

The first two stages (stages M1 and M2) assembles components to the main body of the drawer. The second part of the line (stages M3 and M4) welds the components to the body and performs some final operations including a visual quality check, though parts are not scrapped within the line.

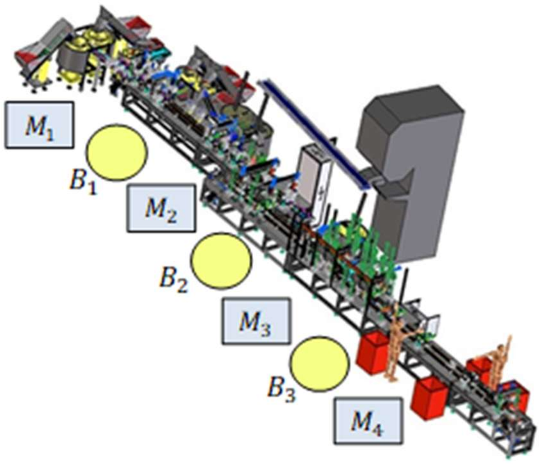


Fig. 8 Multi-stage manufacturing line producing drawers (courtesy of Cosberg).

The improvement actions have been identified by the continuous improvement team, after a careful analysis of the line performance. First, in order to reduce the processing time of the machine 2, which represents the bottleneck machine from the viewpoint of the production speed, it is possible to replace all the pneumatic screwdrivers by automatic electro screwdrivers with torque control as well as replacing the pneumatic drives used to clinching metals sheets by electromechanical servo drive systems that are faster (Improvement action 1 and 2). Also, a bigger improvement on the speed of the same machine can be obtained from improving the product feeders, and increasing the pace of the conveyors on before and after the execution of the tasks themselves (Improvement action 3). These actions require more time, as they need a more deep evaluation and design to guarantee than the final product meets the requirements of quality defined by the company. A general improvement concerning all the failures modes of the second machine could be achieve by incorporating a sensor that check the position of each product on the conveyor, avoiding wrong fitting among the tools and the products (Improvement action 4). This can increment the mean time to failure of all the failures modes at the same time. In a similar manner, on machine 1 also little improvements on the processing times by replacing pneumatic devices by electric ones, are possible (Improvement action 5). In the same logic, improving the speed of feeder technologies using new generational ones and speeding up the transport of the products inside the machine (for example by replacing the electric engine), bigger reduction of the processing times are obtained (Improvement action 6). Other minor improvements can be implemented on machine 1 by acting on the positioning of components, to make them more available and easily replaceable (Improvement actions 7 and 8). In conclusion, 8 improvement actions are identified, meaning that a total of 40320 permutations should be evaluated.

6.2 Results

The proposed methodology has been applied to the case study, and it is also compared to other methods. In particular, the proposed work is compared with the Bottleneck Release, and also with the worst sequence, evaluated by means of exhaustive search. Then, the percentage difference between the cumulative production quantity P obtained with the Proposed work and the Bottleneck Release, with respect to the Worst Case, has been computed and it is shown in Table 2.

The optimal sequence identified by the proposed method guarantees an improvement of more than 7% with respect to the worst sequence that could be implemented. It should be noticed that the optimization and the Bottleneck Release method have some communalities, i.e. the last three actions are the same, as well as action 3 and 4. However, these minor changes affect the effectiveness of the solution.

Table 1 Optimization results of the industrial case-study.

Method	Resulting Sequence	ΔP
Proposed work	$2 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8$	+7.1%
Bottleneck Release	$1 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 6 \rightarrow 8 \rightarrow 7$	+5%
Worst case	$6 \rightarrow 7 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 8$	0

7 Conclusion and future research

In this work, a novel optimization methods for the sequencing of improvement actions in automated multi-stage manufacturing systems is presented. The integration of fast performance evaluation models, that is the stochastic analytical model used in this work, and dynamic programming guarantees to solve a practical problem which implies usually high complexity. The main intuition that allowed the problem solution is that, even if the objective is the sequencing of improvement actions to maximize the system performance in term of cumulated production, at each decision instant the system performance depend on the set of implemented actions but do not depend on the implementation sequence. Results show the effectiveness of this approach in solving complex situations, e.g. when improvement actions differ in effect and implementation time. It is also shown that, despite current manufacturing strategies as those based on the bottleneck identification, no a priori strategy leads to optimal sequences of improvement actions.

Indeed, this work paves the way for the introduction of advanced model-based decision support tools in manufacturing companies.

References

1. Chiang, S. Y., Kuo, C. T., & Meerkov, S. M. (2000). DT-bottlenecks in serial production lines: theory and application. IEEE Transactions on Robotics and Automation, 16(5), 567-580.
2. Chiang, S. Y., Kuo, C. T., & Meerkov, S. M. (2001). c-Bottlenecks in serial production lines: identification and application. Mathematical problems in engineering, 7(6), 543-578.
3. Roser, C., Nakano, M., & Tanaka, M. (2001, December). A practical bottleneck detection method. In Proceeding of the 2001 Winter Simulation Conference (Cat. No. 01CH37304) (Vol. 2, pp. 949-953). IEEE.
4. Grünberg, T. (2004). Performance improvement: Towards a method for finding and prioritising potential performance improvement areas in manufacturing operations. International Journal of Productivity and Performance Management.
5. Tam, A. S., & Price, J. W. (2008). A maintenance prioritisation approach to maximise return on investment subject to time and budget constraints. Journal of Quality in Maintenance Engineering.
6. Lin, L. C., Li, T. S., & Kiang, J. P. (2009). A continual improvement framework with integration of CMMI and six-sigma model for auto industry. Quality and Reliability Engineering International, 25(5), 551-569.
7. Barad, M., & Bennett, G. (1996). Optimal yield improvement in multi-stage manufacturing systems. European Journal of Operational Research, 95(3), 549-565.
8. Kang, Ningxuan, Cong Zhao, Jingshan Li, and John A. Horst. "A Hierarchical structure of key performance indicators for operation management and continuous improvement in production systems." International Journal of Production Research 54, no. 21 (2016): 6333-6350.
9. Stricker, Nicole, Fabio Echsler Minguillon, and Gisela Lanza. "Selecting key performance indicators for production with a linear programming approach." International Journal of Production Research 55, no. 19 (2017): 5537-5549.
10. Colledani, M., Tolio, T., & Yemane, A. (2018). Production quality improvement during manufacturing systems ramp-up. CIRP Journal of Manufacturing Science and Technology, 23, 197-206.
11. Schmitt, R., Heine, I., Jiang, R., Giedziella, F., Basse, F., Voet, H., & Lu, S. (2018). On the future of ramp-up management. CIRP Journal of Manufacturing Science and Technology, 23, 217-225.
12. Magnanini, M. C., Terkaj, W., & Tolio, T. (2021). Robust optimization of manufacturing systems flexibility. Procedia CIRP, 96, 63-68.
13. Colledani, M., Magnanini, M. C., & Tolio, T. (2018). Impact of opportunistic maintenance on manufacturing system performance. CIRP Annals, 67(1), 499-502.