Effective dynamic selection of smart products scheduling rules in FMS

Wassim Bouazza, Djamila Hamdadou, Yves Sallez, Damien Trentesaux

Manufacturing Letters 20 (2019) 45–48

Effective dynamic selection of smart products scheduling rules in FMS

Wassim Bouazza [a,*], Djamila Hamdadou [a], Yves Sallez [b], Damien Trentesaux [b]

[a] LIO, Department of Computer Science, Faculty of Exact and Applied Science, University Oran 1 Ahmed Ben Bella, Oran, Algeria
[b] LAMIH UMR CNRS 8201, Université Polytechnique Hauts-de-France, France

Keywords:

Smart products
Scheduling
Hyper-Heuristic
Distributed control
FMS

A B S T R A C T

The present paper addresses the dynamic scheduling problem of smart products in a Flexible Manufacturing System (FMS). Unrelated machines, multiple product families, operation setups and dynamic job arrivals are considered. The goal is to propose a dynamic selection of smart products scheduling rules enriched with a context perception capability that is optimized using a hyper heuristic. A proof of concept with preliminary results is given.

1. Background and motivation

The future of manufacturing systems is definitely linked to improvements in artificial intelligence. Those evolutions are sustained by numerous technologies like Internet of Things or machine learning. On the one side, constraints due to unrelated machines, different product families, setup operations, and dynamic unpredictable events etc. harden the scheduling of manufacturing tasks. On the other side, new paradigms like Cyber-Physical Production Systems enable resources and products to become more intelligent. Indeed, by communicating and interacting autonomously, they become more active during the scheduling process. Thus, the approaches based on smart products [1,2], have shown the interest of bringing the decisional abilities to products. This distribution may enhance the performance and the reactivity during the scheduling [3,4].

This paper addresses the effective dynamic selection of smart products scheduling rules in Flexible Manufacturing System (FMS) under different original constraints such as unrelated machines, multiple product families, operation setups and dynamic arrivals of product demands. Thus, this paper assumes that a set of smart products (SmP) uses scheduling rules to decide dynamically both assignment and sequencing of jobs. These rules are selected dynamically as well. For that purpose, a Hyper-Heuristic uses a set of descriptive features to automatically characterize each decisional context. The goal is to enhance the relevance of the rules selected by autonomous SmP enriched with a context perception capability. A main Genetic Algorithm (GA) uses parallel multi-agent simulations to evaluate the behaviors of SmP. It can thus operate simultaneously on the context characterization and the scheduling rules to be triggered. A proof of concept on FMS with a comparative results with 16 combinations of classical scheduling rules is detailed.

2. Related works to dynamic rules selection

Scheduling rules like Dispatching Rules (DR) are well known in the literature for their reactivity and their ease of implement [5]. Nevertheless, they may also provide a poor global performance due to the real-time local decisions i.e. myopia effect mainly [6,7]. To counterbalance this poor overall performance, DR can be specifically adapted to the characteristics of the scheduling problem e.g. setup-oriented rules explicitly considering setup operations [8,9]. Nonetheless, if specialized DR can improve the global performance, they reach some limits [10]. Indeed, the lack of genericity makes them unable to deal easily with complex environments with multiple decisional situations.

Recent approaches have been provided through the combination of several DR. In the literature, those methodologies operating on a search space composed of heuristics [11] are named "Hyper-Heuristics". They are defined as heuristics to choose other heuristics [12] or more globally, as heuristics that "manage" other heuristics [13]. According to [14], the distinction is made between perturbative methods considering complete candidate solutions

DOI : 10.1016/j.mfglet.2019.05.004

and modify some of their components [15], and constructive methods considering partial solutions and extend them iteratively [16]. For example in [17], a two-stage Hyper-Heuristic to generate a set of work-centers specific DR is provided. Genetic programming and evolutionary algorithm evolve composite rules while searching for the best ones to be allocated to each work-center. Other works propose more dynamic rule selections by handling the decisional context and trying to switch to the most appropriate DR [18,19]. An example considering explicitly multiple contexts can be found in [20]. A multi-contextual function uses machine idle time and job waiting time to describe the characteristics of a dynamic job shop at a specific time. Results show that the proposed function significatively improves the scheduling performance. In this vein, the authors in [21] define a scheduling context using four characteristics of the manufacturing cell: flexibility, setup-time, operating time and delay. The combination of SmP with Reinforcement Learning method makes the system switching efficiently between DR.

Thus, Hyper-Heuristics may allow to take advantage of fast-reactive scheduling rules by switching among different ones. If some works consider multiple decisional contexts, none really provides a process to automatically generate such contextualization.

3. Proposed approach

3.1. Manufacturing scheduling context

We define the Manufacturing Scheduling Context (MSC) as the characterization of the state of the manufacturing environment, that must be considered in the decision-making process for assignment and/or sequencing of jobs and can, more globally, improve the behaviors of the manufacturing system.

Thereby, two different types of decisions are considered i.e. assignment and sequencing decisions [22,23]. Thus, descriptive features related to the environment, holding products and resources, can be used to determine the MSC. Furthermore, in the field of machine learning, a feature is a measurable property or characteristic of a phenomenon being observed. The choice of those informative, discriminating and independent features is then crucial to get efficient algorithms [24]. To build a such contextual perception, two main elements are used (Fig. 1):

- Basic features: which are values describing the products i.e. jobs and operations, and the resources i.e. cells and machines. Each product has a corresponding job "*j*" and an ordered list of operations "*o*" to achieve and each resource "*r*" belongs to a cell "*i*". In addition, some features may depend on both product and resource e.g. processing times of a given operation in case of unrelated machines.
- Operations: which are used to calculate composite features combining sets of basic features e.g. average value of product families queued in a given cell.

The contextual features (basic feature or composite) are divided into intervals. Each information thus obtained describes a particular aspect of the decisional environment. The smart product strategy (SPS) is composed of two distinct strategies for rules selections during assignment and sequencing (Fig. 2). Each strategy is then a finite number of MSC resulting of the combination of features intervals where each MSC has a specific Assignment Rule (AR) or Sequencing Rule (SR) to be triggered.

To enable the SmP to perceive the MSC and select dynamically the appropriate DR, an SmP architecture, composed of three modules (Fig. 3) is proposed:

- Communication module: manages I/O to permits the SmP to communicate and interact with its environment.
- Decisional module: process incoming inputs to a clearly identified MSC, then triggers the corresponding rule.
- Action module: decomposes the selected rule to a set of actions (e.g. send instructions to a resource).

3.2. Automatic contexts discretization algorithm

The automatic contexts discretization is a multi-step algorithm depicted in Fig. 4. A main GA manage the global process of the Hyper-Heuristic while fast parallel multi-agent simulations are launched to evaluate each SPS. The three main steps are:

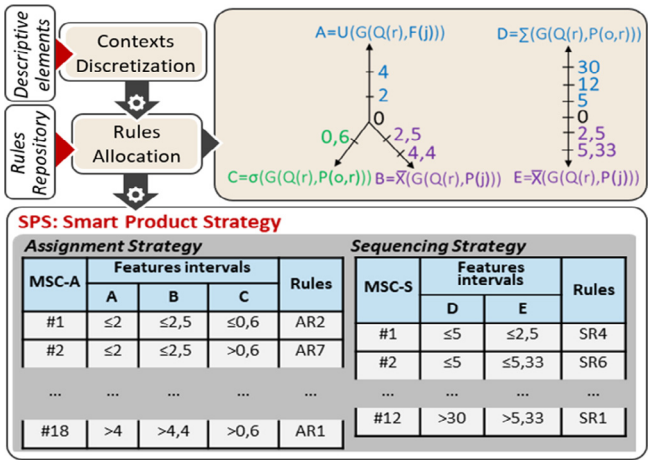

Fig. 2. Process for building SPS.

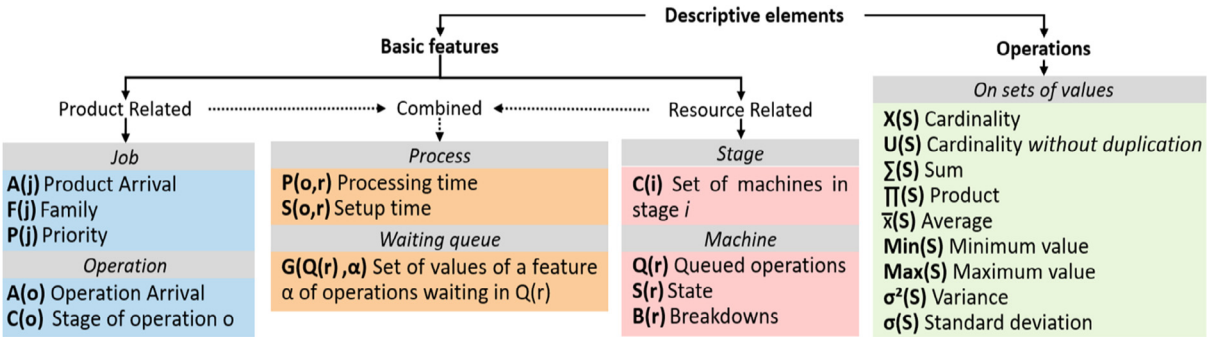

Fig. 1. Basic features and operations used to characterize the decisional context.

DOI : 10.1016/j.mfglet.2019.05.004

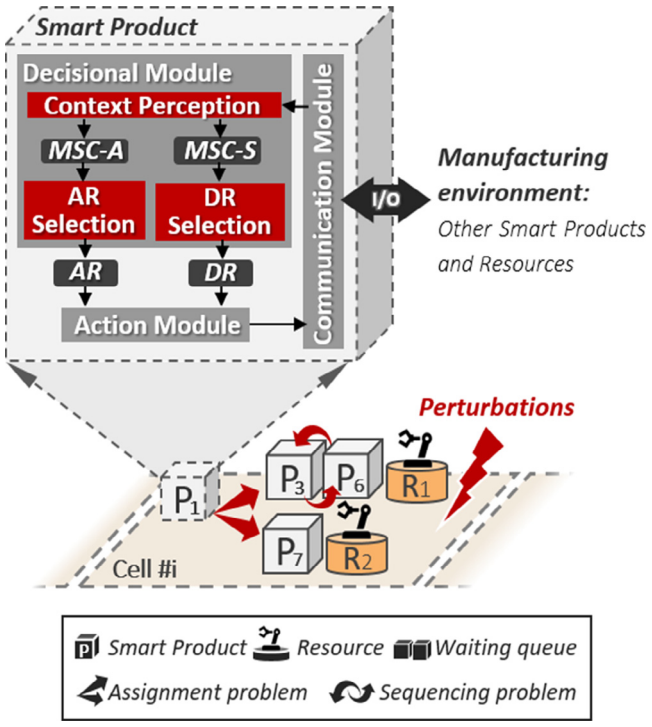

Fig. 3. Architecture of SmP with context perception.

(1) First, an initial population of SPS is generated using the contextual features and operations.
(2) Second, parallel multi-agent simulations are driven to evaluate the performance of each SPS. The simulation parameters are loaded describing resources characteristics and cells organization. Then SmP are launched according to the dynamic scenario until all the manufacturing operations are accomplished.
(3) Third, if the ending condition is not reached yet, SPS are crossed and mutated to generate the next generation.

4. Proof of concept

4.1. Experimentation inputs

To get a preliminary validation, simulations were made on a multi-stage FMS inspired from the pharmaceutical industry and developed using JADE (Java Agent DEvelopment framework).

- FMS: It is composed of 9 resources organized in 4 manufacturing cells. The processing and setup times may differ from a resource to another (Fig. 4c).
- Products: The scenario contains 500 products arriving dynamically following a uniform probabilistic rule during a 5 h period. Each product belongs to one of the 3 products families which determines the main constraints: number and order operations and the processing and the setup times.
- Assignment Rules: LQE (Less Queued Elements), STPT (Shortest Total Processing Time), QPT (Quickest Processing Time) and QST (Quickest Setup Time).
- Sequencing Rules: FIFO (First In First Out), FASFO (First At Stage In First Out), HPF (Highest Priority First) and SJF (Shortest Job First).
- GA: The population contains 500 SPS. The ending condition is set to 100 iterations of the main GA.

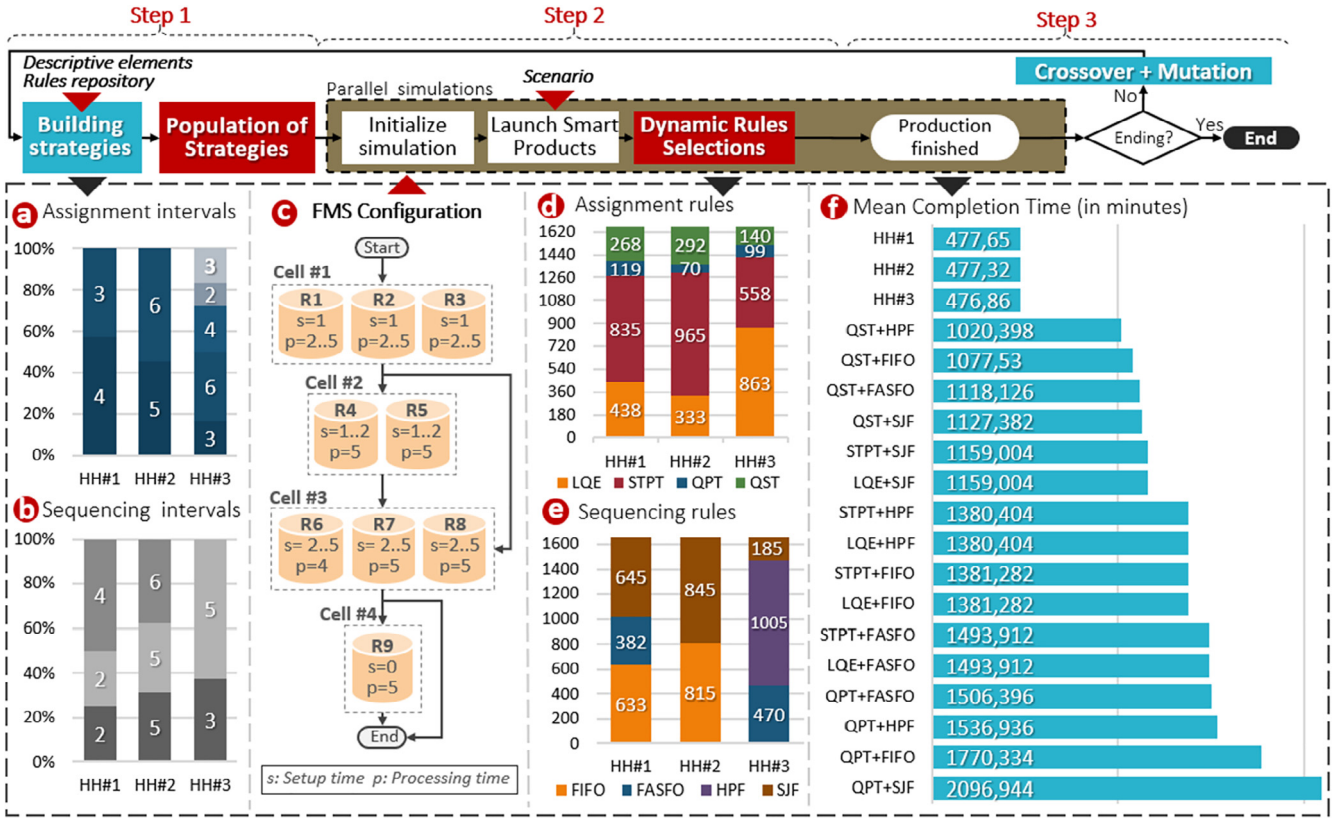

Fig. 4. Experienced FMS and results.

DOI : 10.1016/j.mfglet.2019.05.004

4.2. Results

The chosen objective function is the Mean Completion Time (MCT) [25]. The Hyper-Heuristic proposed being a non-deterministic algorithm, three different executions are made, denoted HH#i. For the 1660 operations composing, 12, 30 and 432 MSC are obtained for the assignment (Fig. 4a) and 16, 150 and 15 MSC are obtained for the sequencing (Fig. 4b).

The SmP switching between 4 AR (Fig. 4d) and 4 SR (Fig. 4e) outperforms the best scheduling rules combination QST + HPF enhancing significatively the MCT (Fig. 4f).

5. Conclusion

The present paper proposed an effective dynamic selection of smart products scheduling rules in FMS under original constraints multiple product families. For that purpose, a Hyper-Heuristic that uses descriptive features to automatically discretize the contextual information's space was proposed. A proof of concept showed the interest of autonomous smart products switching efficiently between different scheduling rules.

Works are already undergoing to provide a larger comparative study with other Hyper-Heuristics and on different configurations of manufacturing plants.

Declaration of Competing Interest

None.

References

[1] Feldhütter V, Steck C, Hawer S, Ten Hompel M. Impacts of product-driven complexity on the success of logistics in the automotive sector. Procedia CIRP 2017;62:129–34.

[2] Borangiu T, Raileanu S, Trentesaux D, et al. Distributed manufacturing control with extended CNP interaction of intelligent products. J Intell Manuf 2014;25:1065–75.

[3] Bouazza W, Sallez Y, Beldjilali B. A distributed approach solving partially flexible job-shop scheduling problem with a Q-learning effect. In: IFAC-PapersOnLine. Elsevier; 2017. p. 15890–5.

[4] Sallez Y. The augmentation concept: how to make a product "active" during its life cycle. In: Borangiu T, Thomas A, Trentesaux D, editors. Studies in Computational Intelligence. Springer; 2012. p. 35–48.

[5] Branke J, Nguyen S, Pickardt CW, Zhang M. Automated design of production scheduling heuristics: a review. IEEE Trans Evol Comput 2016;20:110–24.

[6] Blunck H, Bendul J. Controlling myopic behavior in distributed production systems – a classification of design choices. Procedia CIRP 2016;57:158–63.

[7] Zambrano Rey G, Pach C, Aissani N, et al. The control of myopic behavior in semi-heterarchical production systems: a holonic framework. Eng Appl Artif Intell 2013;26:800–17.

[8] Pickardt CW, Branke J. Setup-oriented dispatching rules – a survey. Int J Prod Res 2012;50:5823–42.

[9] Gavett J. Three heuristic rules for sequencing jobs to a single production facility. Manage Sci 1965;11:166–76.

[10] Sarin SC, Varadarajan A, Wang L. A survey of dispatching rules for operational control in wafer fabrication. Prod Plan Control 2011;22:4–24.

[11] Swiercz A. Hyper-heuristics and metaheuristics for selected bio-inspired combinatorial optimization problems. Heuristics and Hyper-Heuristics – Principles and Applications. InTech; 2017.

[12] Ochoa G, Qu R, Burke EK. Analyzing the landscape of a graph based hyper-heuristic for timetabling problems. In: Proc 11th Annu Conf Genet Evol Comput – GECCO '09; 2009. p. 341–8.

[13] Vázquez Rodríguez JA, Petrovic S, Salhi A. A combined meta-heuristic with hyper-heuristic approach to the scheduling of the hybrid flow shop with sequence dependent setup times and uniform machines. In: Proc 3rd Multidiscip Int Conf Sched Theory Appl; 2007. p. 506–513.

[14] Hoos HH. Stochastic local search – method, models and applications. Morgan Kaufmann Publishers; 1998.

[15] Cano-Belmán J, Ríos-Mercado RZ, Bautista J. A scatter search based hyper-heuristic for sequencing a mixed-model assembly line. J Heuristics 2010;16:749–70.

[16] Mascia F, López-Ibáñez M, Dubois-Lacoste J, Stützle T. From grammars to parameters: automatic iterated greedy design for the permutation flow-shop problem with weighted tardiness. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics) 2013:321–34. 7997 LNCS.

[17] Pickardt CW, Hildebrandt T, Branke U, et al. Evolutionary generation of dispatching rule sets for complex dynamic scheduling problems. Int J Prod Econ 2013;145:67–77.

[18] Liu M, Wu C. Genetic algorithm using sequence rule chain for multi-objective optimization in re-entrant micro-electronic production line. Robot Comput Integr Manuf 2004;20:225–36.

[19] Pierreval H, Mebarki N. Dynamic scheduling selection of dispatching rules for manufacturing system. Int J Prod Res 1997;35:1575–91.

[20] Lu MS, Romanowski R. Multicontextual dispatching rules for job shops with dynamic job arrival. Int J Adv Manuf Technol 2013;67:19–33.

[21] Bouazza W, Sallez Y, Aissani N, Beldjilali B. A model for manufacturing scheduling optimization through learning intelligent products. In: Studies in Computational Intelligence. Springer International Publishing; 2015. p. 233–41.

[22] Harjunkoski I, Maravelias CT, Bongers P, et al. Scope for industrial applications of production scheduling models and solution methods. Comput Chem Eng 2014;62:161–93.

[23] Trentesaux D, Pach C, Bekrar A, et al. Benchmarking flexible job-shop scheduling and control systems. Control Eng Pract 2013;21:1204–25.

[24] Bishop CM. Pattern Recognition and Machine Learning; 2007.

[25] Aydilek H, Allahverdi A. Heuristics for no-wait flowshops with makespan subject to mean completion time. Appl Math Comput 2012;219:351–9.