Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 79 (2019) 233–238

www.elsevier.com/locate/procedia

12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy

Intelligent cloud manufacturing platform for efficient resource sharing in smart manufacturing networks

Alessandro Simeone[a,*], Alessandra Caggiano[b,c], Lev Boun[d], Bin Deng[a]

[a]*Intelligent Manufacturing Key Laboratory of Ministry of Education, Shantou University, 515063, Shantou, China*
[b]*Department of Industrial Engineering, University of Naples Federico II, 80125, Naples, Italy*
[c]*Fraunhofer Joint Laboratory of Excellence on Advanced Production Technology (Fh-J_LEAPT UniNaples), 80125, Naples, Italy*
[d]*Dynamic Computer Systems, PoB 1341, 30300, Atlit, Israel*

* Corresponding author. Tel.: +8613411903575; *E-mail address:* simeone@stu.edu.cn

Abstract

Modern manufacturing demands are characterized by high fluctuations with negative impact on resource efficiency. In this framework, Industry 4.0 key enabling technologies such as cloud manufacturing enable the sharing of distributed resources for effective use at industrial network level. In this work, an intelligent cloud manufacturing platform is proposed to increase resource efficiency in a manufacturing network through dynamic sharing of manufacturing services, including computational, software as well as physical manufacturing resources, that can be offered on demand according to a service-oriented paradigm. The cloud-based platform includes a database module where user input data are collected, an intelligent module for data processing, optimization and feasible solutions generation, and a decision support module for solutions evaluation and comparison. A case study demonstrates technical and economic advantages for industrial resource efficiency improvement.

Peer-review under responsibility of the scientific committee of the 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering.

Keywords: Cloud manufacturing; Industry 4.0; Resource efficiency; Smart manufacturing network

1. Introduction

Modern manufacturing industry is exposed to high competitiveness at global scale and largely variable demands, which negatively affect manufacturing resource efficiency.

The efficient use of manufacturing resources can be effectively promoted by Industry 4.0 key enabling technologies (KETs) like Cloud Manufacturing (CM). The latter represents an extension of cloud computing with the aim to provide on-demand manufacturing services to several industrial users according to a service-oriented paradigm [1, 2]. At the industrial network level, the full-scale sharing and on-demand-use of distributed computational, software, digital and physical manufacturing resources via cloud manufacturing allows to create smart manufacturing networks with improved resource efficiency, higher productivity and utilization rates [2, 3].

Through the cloud, users can get ubiquitous access to smart machines, production systems, as well as big amounts of data generated by different sources, including other users, sensor systems and intelligent computation [3].

However, to realize the cloud manufacturing paradigm, there are still open research gaps including the development of cloud-based platforms, proper interfaces for production systems/users, service-oriented delivery of automation utilities.

In the recent literature, innovative cloud manufacturing solutions have been proposed for a wide range of applications such as production planning, monitoring, control, management and design [4–6]. In [7], based on the use of sensor networks on the shop floor, information fusion technique and data communication through the internet, a cloud-based approach is employed for assigning jobs to the available CNC machines. A service-oriented cloud-based approach is proposed in [8] for adaptive process planning based on the collection and

Peer-review under responsibility of the scientific committee of the 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering.
10.1016/j.procir.2019.02.056

processing of data on machine tools status, specifications and availability. In [9], a web-based service-oriented system is developed for distributed machining process planning in a decentralized and dynamic manufacturing environment characterized by unpredictable shop-floor changes and network-wide accessibility to manufacturing services.

In the present work, an intelligent cloud manufacturing platform is proposed to realize a smart manufacturing network with increased resource efficiency, allowing the dynamic sharing of manufacturing services. The latter can be offered on demand according to a service-oriented paradigm, so that convenient sharing of a variety of distributed manufacturing resources is realized in a dynamic way based on the actual user needs. In this framework, different users are enabled to search and request from the manufacturing cloud the services needed to fulfill the required manufacturing tasks and dynamically assemble them into a manufacturing service solution. Via cloud, both physical and computational resources are offered in the network, including physical machine tools, data storage and intelligent computation algorithms for decision-making.

The employment of such cloud manufacturing platform represents an advantage for manufacturing service customers, that will get access to a wider resource network with many solutions for their tasks, as well as for manufacturing service suppliers, that will be able to get new jobs and increase their resource efficiency. A case study involving the implementation of the cloud manufacturing platform to sheet metal cutting services is employed to demonstrate the technical and economic advantages for industrial resource efficiency improvement.

2. Cloud Manufacturing Framework

The cloud manufacturing framework developed in this research work to enable the sharing of distributed resources for effective use at industrial network level is illustrated in Fig. 1.

Distributed manufacturing resources offered by different suppliers within the manufacturing network are dynamically shared with users connected via Internet to the cloud platform.

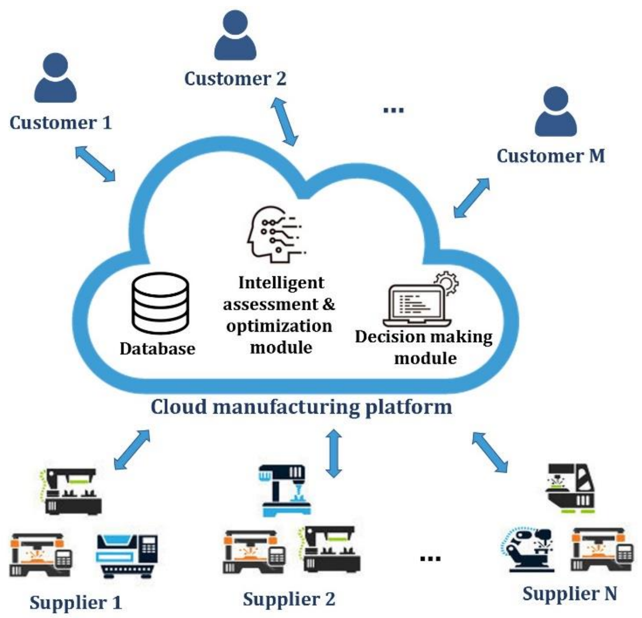

Fig. 1. Cloud manufacturing framework for smart manufacturing networks.

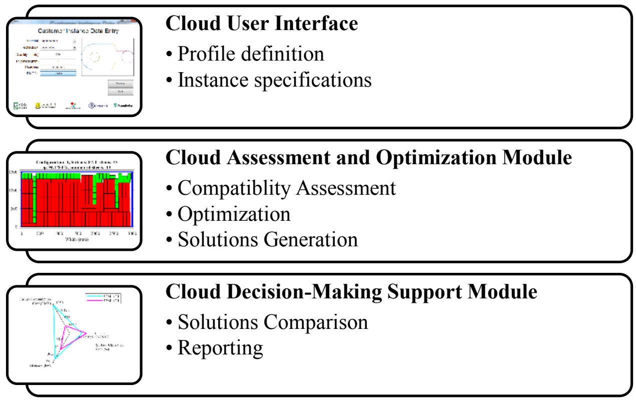

Fig. 2. Cloud platform modules and tasks.

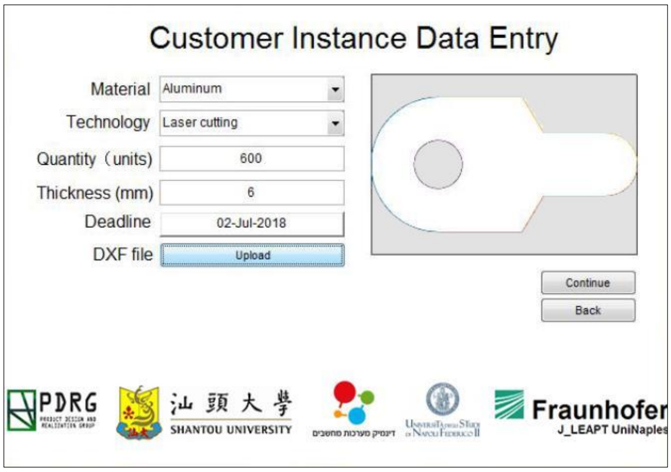

Fig. 3. Customer instance data entry GUI.

The cloud platform collects the manufacturing services requests from customers and the offers from suppliers and processes them via intelligent decision-making algorithms to find the best manufacturing solution according to the user needs. To perform these tasks, the platform includes a database module where user input data are collected, an intelligent module for data processing, optimization and feasible solutions generation, and a decision support module for solutions evaluation and comparison (Fig. 1).

To collect the information from the users into the database, a cloud user interface is provided. The tasks carried out at the cloud user interface level, the cloud compatibility assessment and optimization module and the cloud decision-making support module level are illustrated in Fig. 2. The cloud platform presented here refers to sheet metal cutting services.

3. Cloud User Interface

A graphic user interface (GUI) is created to enable users to create a personal profile and entry instance data into the cloud platform. Two types of instances can be provided to the cloud: customer instance and supplier instance. A screenshot of the GUI for customer instance data entry is shown in Fig. 3.

In the proposed system, based on cloud technology, each user will have a "personal" space secured by a personal and unique certificate (SSL). The certificate will be created during the account process and the certificate key will be sent to the user automatically and not kept on the system. If the key gets lost, no one will have access to the data. The system is characterized by a double authentication, i.e. Certificate, Login/Password and Confirmation code. From the personal space, all aggregated information will be transferred to the system according to an internal ID number with encrypted user personal details (e.g. location, technology, part details, etc.).

3.1. Customer Instance

A generic customer instance is made of the following fields:

$$C_i = \{ID_{ci}, Q_{ci}, t_{ci}, m_{ci}, T_{ci}, DL_{ci}, CAD_{ci}, L_{ci}\} \tag{1}$$

where ID_{ci} is the unique instance identification number, generated automatically by the system, Q_{ci} is the batch quantity (units), t_{ci} is the thickness (mm), m_{ci} is the material (chosen from a drop-down list), T_{ci} is the cutting technology (from a drop-down list, with the option of choosing none), DL_{ci} is the customer deadline (dd-mmm-yyyy date format), CAD_{ci} is the CAD file (in .dxf format) containing geometrical information (size and shape) as well as tolerances, and L_{ci} is the customer location. The system will select the most severe tolerance:

$$Tol_{ci} = \min_{CAD}(Tol(CAD_{ci})) \tag{2}$$

3.2. Supplier Instance

A generic supplier instance is made of the following fields:

$$S_i = \left\{ID_{sj}, Q_{sj}, M_{sj}, m_{sj}, t_{sj}, A_{sj}, DL_{sj}, CAD_{sj}, L_{sj}\right\} \tag{3}$$

where M_{sj} represents a specific machine tool, including information on the machine model, technology, tolerance, power consumption, cutting parameters, kerf width, availability factor and scrap rate. A_{sj} is the metal sheet size ([Width (W), Height (H)], mm), CAD_{sj} is the CAD file (.dxf format) of supplier Jobs to be carried out by the deadline and L_{sj} is the supplier location.

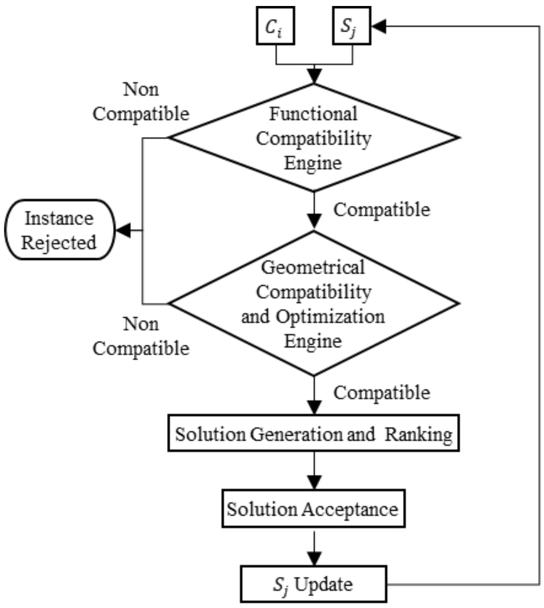

Fig. 4. Compatibility assessment flow chart

4. Cloud Intelligent Assessment and Optimization Module

The objective of the cloud platform is to assess the compatibility of customer and supplier instances and generate a number of suitable solutions using an intelligent optimization algorithm as illustrated in Fig. 4. In this section, the functional and geometrical compatibility and optimization engines are discussed, and the various cost functions are presented.

4.1. Functional compatibility Engine

Customer and supplier instances created via cloud user interface are subject to a first matching procedure to evaluate their compatibility. The Functional Compatibility indicator FC_{ij} between a single Customer Instance C_i and a single Supplier Instance S_j is a Boolean variable defined as:

$$FC_{ij} = \begin{cases} 0 & \text{Non compatible} \\ 1 & \text{Compatible} \end{cases} \tag{4}$$

The functional compatibility is assessed by verifying a number of conditions to be satisfied simultaneously:

- Deadline: the supplier deadline should not succeed the customer deadline: $if\ DL_{si} \succ DL_{ci} \rightarrow FC_{ij} = 0$
- Material and thickness: the customer instance material and thickness should be equal to the supplier ones: $if\ m_{ci} \vee t_{ci} \neq m_{sj} \vee t_{sj} \rightarrow FC_{ij} = 0$
- Technology: when specified by the customer, it should match exactly the technology offered by the supplier: $if\ T_{ci} \neq T(M_{sj}) \rightarrow FC_{ij} = 0$
- Tolerances: the technological characteristics of the Supplier instance should satisfy the tolerances defined by the Customer: $if\ Tol_{sj} > Tol_{ci} \rightarrow FC_{ij} = 0$

When the customer does not specify the technology ($T_{ci} = \emptyset$), a technology compatibility assessment is carried out using materials, thickness and tolerances. In all other cases $FC_{ij} = 1$.

4.2. Geometrical compatibility and optimization Engine

Once the functional compatibility has been verified, the compatible pairs of instances are inputted to the Geometrical compatibility and optimization engine. In this step, the objective is to generate a number of suitable solutions based on the geometrical requirements. This procedure is configurable as a cutting stock problem (CSP) [10], where the problem input is given by a set of item sizes (products) and demands, and by a set of master metal sheets of given dimensions. The task is to define a cutting pattern in order to minimize the total number of master metal sheets used to produce the required items [11].

The cloud platform utilizes a Genetic Algorithm (GA) based optimization system [12, 13]. The genetic algorithm aims at finding a solution in terms of number of items to be placed in each metal sheet and their rotation. The items are subsequently arranged within the metal sheet so as to avoid fragmenting the available remaining surface to allow for a better utilization [14].

In this respect, the adopted strategy is the finite bottom-left (FBL) [15]. The algorithm initially sorts the items by nonincreasing width. The current item is then packed in the lowest position of any initialized bin, left justified; if no bin can accommodate it, a new one is initialized [16].

Additional constraints are applied to the items spacing, specifically: the distance between the metal sheet edge and each item is equal to the kerf width related to the machine in consideration. The space between two items is equal to 2 x kerf width (the kerf width is provided as a machine tool attribute).

If the GA is not able to find a solution, it means that the current Customer Instance C_i doesn't have enough available surface for the current Supplier Instance S_j, hence the instances are not compatible. If the GA converges to a solution, it will generate a number of geometrical configurations of metal sheets that allow for the realization of the customer batch of items C_i within the supplier batch S_j.

Each generated configuration is characterized by a Surface Utilization Rate, η_{ij}, defined as the ratio of the utilized area and the total metal sheets area, as reported in Eq. 5

$$\eta_{ij} = \frac{\sum\left(Area\left(CAD_{sj}\right) + Area\left(CAD_{ci}\right)\right)}{\sum A_{sj} \times number\ of\ sheets} \tag{5}$$

The Surface Utilization Rate should be maximized to reduce the scrap surface related to the metal cutting instance. Solutions are ranked in a descending order according to η_{ij}.

A limitation to this approach is represented by the locality of the optimum solution [11]. To tackle this issue, it is possible to empirically set the algorithm parameters as well as adopt a parallel algorithm strategy [17, 18].

4.3. Energy consumption cost function

The energy consumption cost E_{ij} is calculated based on unitary energy cost, machine average power and cutting time.

To estimate the cutting time, the cutting speed, v_c, is calculated based on the machine tool type. As regards the laser cutting speed, it is computed on the basis of the experimental curves, available from the database, which are a power function of the workpiece thickness, expressed as follows [19]:

$$v_c = \alpha {t_{sj}}^{\beta} \tag{6}$$

Where α and β depend on the laser type (e.g. CO_2, Fiber, etc.) and power (e.g. 3000 W, 5000 W, etc.) as well as on the workmaterial (e.g. stainless steel, carbon steel, copper, etc.) and their values will be available in the cloud database.

As regards the waterjet cutting speed, it is calculated based on workpiece material and thickness, water pressure, orifice size, focus tube diameter, abrasive material and rate [19].

The Cutting Time is then computed as the ratio of the total cutting length, i.e. the length of the path required to cut the part geometry, and the cutting speed, taking into account the machine availability factor as shown in Eq. 7.

$$Cutting\ Time = \frac{Total\ Cutting\ Length}{Availability\ \times v_c} \tag{7}$$

Hence, the energy consumption is calculated by multiplying the average machine power by the cutting time, and the final energy consumption cost is obtained by multiplying the energy consumption by the local unitary energy cost.

$$E_{ij} = Unitary\ Energy\ Cost \times Energy\ cons. \times Cutting\ Time \tag{8}$$

4.4. Distance cost function

The transportation costs can be modelled by introducing a distance related cost D_{ij}, to be calculated for each instance pair based on the customer and supplier locations L_{sj} and L_{ci} (utilizing a distance cost coefficient δ_D). The distance computation, along with an estimation of costs can be obtained using third party modules [20, 21]. Solutions are ranked in a descending order according to D_{ij}.

$$D_{ij} = \delta_D \left\| L_{sj} - L_{ci} \right\| \tag{9}$$

4.5. User rating

Customers and Suppliers can be endowed with a score, respectively R_i and R_j, related to their reputation and historical quality feedback. Each solution can be weighted according to the supplier rating and therefore ranked in descending order.

5. Cloud Decision-Making Support Module

The Cloud Decision-Making Support Module provides final recommendations for users to identify an appropriate match for the realization of the desired products. The compatible solutions generated at the cloud level are ranked according to total utilization rate (%), energy consumption cost (CNY), distance (km) and rating.

Table 1. Case study suppliers table

		Suppliers															
		S_1								S_2			S_3				
Model		**TruLaser 3030 3200 W**			**Flow Mach 500**					**TruLaser 3040 4000 W**			**TruLaser 3030 fiber 4000 W**				
Technology		Laser			Waterjet					Laser			Laser				
Laser Type		CO_2			-					CO_2			Fiber				
Laser Power (W)		3200 W			-					4000 W			4000 W				
Max Dimension (mm)		1500 x 3000			1500 x 3000					2000 x 4000			1500 x 3000				
Power (kW)		29			35					31			14				
Materials		SST	CS	Al	SST	CS	Al	Cu	Brass	SST	CS	Al	SST	CS	Al	Cu	Brass
Thickness (mm)	**Min**	0.5	0.5	0.5	2.5	2.5	2.5	2.5	2.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
	Max	12.7	20	8	300	300	300	300	300	26	20	10	20	25	20	8	8
Kerf width (mm)		1			1.2					1			1				
Availability (%)		90			90					90			90				
Scrap rate (%)		5			5					5			5				

The solutions report is organized in a summary table for ease of use, where the rows represent the compatible solutions and the columns represent the various parameters listed above.

The users will select the best solution according to their requirements and an instance match is created upon the agreement by both parts on price and delivery details.

The outcome will be sent directly to the cloud user according to a preferred path (i.e. email, SMS or Web Service).

5.1. Instance agreement and new users

Once both users have agreed on an instance match, the platform will update S_j with the inclusion of the CAD_{ci} data. The updated S_j is then stored in the cloud database.

Starting from the agreement date, and until the deadline date DL_{sj}, there can exist new user(s) who can fit their C_i in the updated S_j as illustrated in the loop in Fig. 3.

In this scenario, the potential final surface utilization rate will be increased given the higher initial η_{ij}, on the other hand, there will be less probability to geometrically compatible as the new available area will be smaller.

6. Case Study

To exemplify the proposed framework, a case study including a single customer and three suppliers is reported.

The machine details specified by the suppliers and stored in the cloud database are reported in Table 1. The allowed metal sheet materials include Stainless Steel (SST), Carbon Steel (CS), Copper (Cu), Aluminum (Al) and Brass. The customer and suppliers' instances are reported in Tab. 2-3, respectively.

Table 2. Customer Instance

ID_{ci},	Q_{ci}	m_{ci}	t_{ci},	T_{ci},	DL_{ci}	L_{ci}	CAD_{ci}
C1	600	Al	6	Laser	28-Jun 2018	Shantou (China)	w = 90 mm h = 160 mm

Table 3. Suppliers Instances

ID_{sj},	Q_{sj}	M_{sj}	m_{sj}	t_{sj},	A_{sj},	DL_{sj}	L_{sj}	CAD_{sj}
S_1M_1	1500	TL3030	Al	8	1500 x 3000	29-Jun 2018	Shenzhen (China)	w = 210 mm h = 297 mm
S_1M_2	2000	FM500	Brass	10	1500 x 3000	10-Jul-2018	Shenzhen (China)	w = 100 mm h = 150 mm
S_2M_1	1000	TL3040	Al	6	2000 x 3000	25-Jun 2018	Jieyang (China)	w = 190 mm h = 180 mm
S_3M_1	1200	TL3030	Al	6	1500 x 3000	21-Jun 2018	Chaozhou (China)	w = 100 mm h = 400 mm

For this case study, the following assumptions were considered:

- All the items are represented by rectangles and no complex geometries are involved.
- All the suppliers have the same rating, to equalize the rating ranking.
- All tolerances defined by the customer are perfectly compatible with the tolerances provided by the suppliers' machines.
- The metal sheets availability is to be considered unlimited for all the suppliers.
- The energy cost was set to 0.8901 CNY/kWh for all the suppliers [22].
- In the deadline formulation, the transportation and delivery time was not taken into account.

With reference to the customer instance C_1, the functional compatibility engine rejected the supplier instance S_1M_1 as the thickness and deadline were not compatible with the customer instance. Similarly, instance S_1M_2 was rejected because of non-compatibility of technology, thickness and deadline. Consequently, the geometrical compatibility and optimization engine was applied only to instances S_2M_1 and S_3M_1.

In this work, the genetic algorithm was configured utilizing a population size of 500, 2 genes (representing the item inclusion and rotation, respectively) and a maximum number of generations equal to 100. The permutation probability was set to 0.5 and the crossover probability was set to 0.5 [23, 24].

The energy consumption cost for all the instance pairs was computed according to Eq. 8. The distance computation was carried out utilizing third party software, namely Baidu Maps © [21]. The results of the platform computation are reported for the two compatible instances in Table 4, in terms of total utilization rate, energy consumption cost and supplier-customer distance.

A useful way to display the solutions generated by the platform is represented by the spider web chart illustrated in Fig. 5 which allows for a simultaneous visual comparison.

Table 4. Table of compatible solutions results

Instance	Total Utilization Rate (%)	Energy Consumption Cost (CNY)	Distance (km)
$M_2S_1+C_1$	91.43	1004	37.0
$M_3S_1+C_1$	92.30	654	35.3

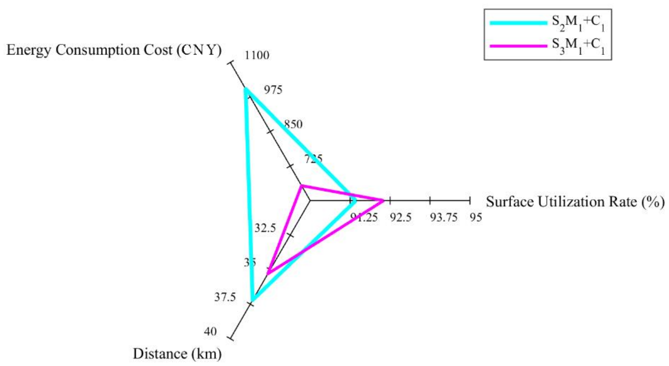

Fig. 5. Compatible solutions comparison

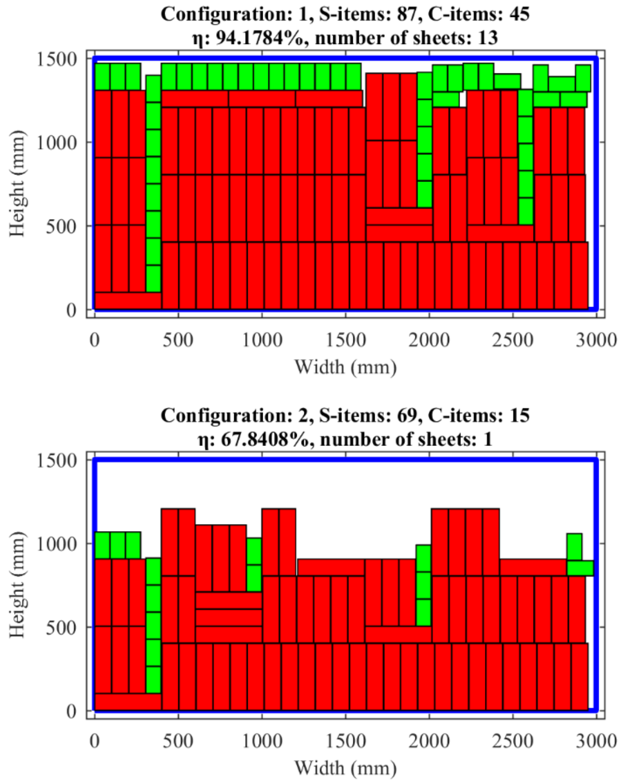

Fig. 6. Metal sheets configurations

Table 5. Utilization rate comparison

Instance	Total Utilization Rate (%)
M_3S_1	84.11
$M_3S_1+C_1$	92.30

The sheets configurations generated by the GA to realize the pair of instances $S_3M_1+C_1$ are reported in Fig. 6. The red items are related to the supplier S_3 and the green items are related to the customer. The computing time for this solution was 159 s.

The first configuration consists of 13 sheets containing 87 supplier items and 45 customer items each, leading to a utilization rate of 94.18%. The second configuration consists of 1 sheet containing 69 supplier items and 15 customer items each, leading to a utilization rate equal to 67.84%. The total surface utilization rate for this pair of instances is 92.30 %.

7. Concluding discussions

This paper proposed an intelligent cloud manufacturing platform to increase computational and physical resource efficiency in a manufacturing network through dynamic sharing of manufacturing services. An interface module for user data input, a cloud-based intelligent module for data processing, optimization and feasible solutions generation, and a decision support module for solutions evaluation and comparison were created. The reported case study related to sheet metal cutting services shows how the cloud framework improved the utilization rate of the supplier resources (Table 5). Assuming that the supplier is endowed with a cutting stock optimization tool, the utilization rate for the batch would be 84.11%. By incorporating the customer instance described above, the total utilization rate rises to 92.30%.

Future work will involve the optimization engine improvement, including handling of complex geometries, addition and modelling of a third type of user (i.e. transporter) and the investigation of alternative intelligent algorithms.

Acknowledgements

The research activities were carried out in the framework of the project CLOUD MODE 'CLOUD Manufacturing for On-Demand manufacturing sErvices' (000011– ALTRI_DR_ 3450_2016_RICERCA_ATENEO-CAGGIANO) funded by the University of Naples Federico II within the 'Programma per il finanziamento della ricerca di Ateneo' (2016–2018).

References

[1] Zhang L, Luo Y, Tao F, et al. Cloud manufacturing: a new manufacturing paradigm. Enterp Inf Syst 2014; 8(2): pp. 167–87.
[2] Xu X. From cloud computing to cloud manufacturing. Robot Comput Integr Manuf 2012; 28(1): pp. 75–86.
[3] Wu D, Rosen DW, Schaefer D. Scalability Planning for Cloud-Based Manufacturing Systems. J Manuf Sci Eng 2015; 137(4):040911.
[4] Caggiano A. Cloud-based manufacturing process monitoring for smart diagnosis services. Int J Comput Integr Manuf 2018.
[5] Caggiano A, Segreto T, Teti R. Cloud Manufacturing Framework for Smart Monitoring of Machining. Procedia CIRP 2016; 55: pp. 248–53.
[6] Caggiano A, Segreto T, Teti R. Cloud Manufacturing on-demand services for holistic quality assurance of manufactured components. Procedia CIRP 2018; 67: pp. 144–49.
[7] Tapoglou N, Mehnen J, Vlachou A, et al. Cloud-Based Platform for Optimal Machining Parameter Selection Based on Function Blocks and Real-Time Monitoring. J Manuf Sci Eng 2015; 137(4):040909.
[8] Mourtzis D, Vlachou E, Xanthopoulos N, et al. Cloud-based adaptive process planning considering availability and capabilities of machine tools. J Manuf Syst The Society of Manufacturing Engineers 2016; 39: pp. 1–8.
[9] Wang L. Machine availability monitoring and machining process planning towards Cloud manufacturing. CIRP J Manuf Sci Technol CIRP 2013; 6(4): pp. 263–73.
[10] Delorme M, Iori M, Martello S. Bin packing and cutting stock problems: Mathematical models and exact algorithms. European Journal of Operational Research 2016[Online] 2016.
[11] Malaguti E, Medina Durán R, Toth P. Approaches to real world two-dimensional cutting problems. Omega (UK) 2014; 47: pp. 99–115.
[12] Cui Y, Zhao Z. Heuristic for the rectangular two-dimensional single stock size cutting stock problem with two-staged patterns. Eur J Oper Res 2013; 231(2): pp. 288–98.
[13] Kallrath J, Rebennack S, Kallrath J, et al. Solving real-world cutting stock-problems in the paper industry: Mathematical approaches, experience and challenges. Eur J Oper Res 2014; 238(1): pp. 374–89.
[14] Dalalah D, Khrais S, Bataineh K. Waste minimization in irregular stock cutting. J Manuf Syst 2014; 33(1): pp. 27–40.
[15] Burke E, Hellier R, Kendall G, et al. A New Bottom-Left-Fill Heuristic Algorithm for the Two- Dimensional Irregular Packing Problem. Oper Res 2006; 54(3).
[16] Lodi A, Martello S, Monaci M, et al. Two-Dimensional Bin Packing Problems. In: Paradigms of Combinatorial Optimization Wiley-Blackwell 2014; pp. 107–29.
[17] Yu B, Yang Z, Sun X, et al. Parallel genetic algorithm in bus route headway optimization. Appl Soft Comput J 2011; 11(8): pp. 5081–91.
[18] León C, Miranda G, Rodríguez C, et al. 2D Cutting Stock Problem: A New Parallel Algorithm and Bounds. Euro-Par 2007 Parallel Processing 2007[Online] Springer Berlin Heidelberg: Berlin, Heidelberg 2007.
[19] Oberg E, D. Jones F, L. Horton H, et al. Machinery ' s Handbook,. Machinery's Handbook 28th Edition 2008[Online] 2008.
[20] Google Maps ©.
[21] Baidu Maps ©.
[22] Unified service platform of China Southern Power Grid.
[23] Goldberg DE. Genetic Algorithms in Search, Optimization and Machine Learning,. Addison-Wesley 1989[Online] 1989.
[24] Gould O, Simeone A, Colwill J, et al. Optimized Assembly Design for Resource Efficient Production in a Multiproduct Manufacturing System. Procedia CIRP 2017[Online] 2017.