Negotiating and Sharing Capacities of Large Additive Manufacturing Networks

Massimo De Falco[1*]
E-mail: mdefalco@unisa.it
Nicola Mastrandrea[1*]
E-mail : nmastrandrea@unisa.it

Luigi Rarità[2*]
E-Mail : lrarita@unisa.it
University Of Salerno

Abdallah Asan Alalawin[3]

University of Dubai
E-Mail : ahalalawin@ud.ac.ae

Abstract

This paper focuses on dynamics of productive and demanding nodes for Scattered Manufacturing Networks within 3D Printings contexts. The various nodes issue orders or sell production slots in order to achieve their own aims. An orchestrator coordinates the dynamics along the network according to principles of sustainability, equated shared resources and transparency by managing communication activities among nodes. In particular, suitable tradeoffs occur by a unique framework that, with the aim of optimizing the overall costs, suggests either logistics paths along the network or negotiation policies among nodes in order to reallocate resources. Numerical examples present the proposed approach.

Keywords: Industry 4.0, Additive Manufacturing, Sharing Capacities, Operation Models, Optimization of networks

JEL Codes: C02; O21 and P40

1. INTRODUCTION

Nowadays, international markets are facing an increasing need for individualized products. Moreover, mass customization creates a growing competition that requires high agility, rapid changes in the customized production style and fast reconfiguration of manufacturing systems.

This is also the result of the emerging Industrial Revolution, namely Industry 4.0 (I40), which is realizing a suitable interconnection among different fields of action within and between organizations (MacDougall, 2014) through the achievement of resource efficiency (Hauder et al. 2017). By connecting organizational units, various and new synergy effects should be detected and used (MacDougall, 2014). Therefore, the approach of I40 indicates a possible focus on the holistic consideration of different issues and problem models of organizations (Hauder et al. 2017).

The realization of I40 is possible due to the implementations of Internet of Things (IoT) and Cyber-Physical Systems (CPSs) (Wan et al. 2016), useful to bridge either the virtual world or the physical one. In particular, following the original idea by Mark Weiser about *ubiquitous computing* (Weiser, 1993) and its reworking in the concept of *Agile Factory*, created by Constantin Scheuermann et. al. (Scheuermann et al. 2015), for the domain of manufacturing, we deal with the context of Advanced Manufacturing Technologies (AMTs) in terms of *Scattered Manufacturing* (SM).

SM allows production everywhere in anytime inside the domains of the networks. Factories can rely on on-demand manufacturing services by shared resource pools in a geographical distributed network.

AMT aims at improving value chains and value-added networks in industry to ensure industrial competitiveness (Durão et al. 2017). Key challenges are the integration of seamless digital workflows throughout the product lifecycle, the development of highly flexible and adaptive manufacturing processes, and the capabilities to manufacture individualized products at the price of mass production (Anderl, 2015).

Production sites are becoming rapidly adaptive while remaining economically productive (Anderl, 2015). This situation enables to decentralize production through autonomous tasks based on Cyber-Physical Systems (Brettel et al. 2014), improved quality control (Ariss et al. 2000, Schroder and Sohal, 1999), and reduced inventories (Jonsson, 2000). It is foreseen that smart factories are networked in cross-company collaboration, sharing relevant resources and information, and thus creating a dynamic production environment. Possible advantages are evident, such as an increased product variety, reduced lifecycle times and geographically distributed markets brought by a dynamic environment (Durão et al. 2017).

In order to raise the perception of the value for the customer, one strategy that can be adopted is the geographically distributed manufacturing, transferring the production closer to the client and reducing the time to deliver products (Khajavi et al. 2014). However, SM by adoption of AMTs is still in its early research phase. Key challenges are represented by improvement and standardization of the information flow, communication, and control among production nodes, changing the overall work organization structures (Schroder and Sohal, 1999).

Much of the shift towards I40, indeed, is driven by the emergence of 'Big-Data' and the issues, connected to the ways by which industrial operations collect, manage and interpret their data, remain prevalent (Chen et al. 2015). Considerations about Big Data and the treatment of large datasets are an intrinsic challenge of each system operating in an I40 environment. In fact, traditional statistical processing methods are often useless due to the complexity and the sheer size of large datasets (Oliff and Liu, 2017). Current implementations have demonstrated an adaptive scheduling, a real-time modelling of processes, and Decision Support Systems, used to refine processes and component design (Oliff and Liu, 2017).

For the optimization of issues within the context of production and logistics, a typical aim is gaining quantitative improvements, which also correspond to an increase of resource efficiency (Hauder et al. 2017). Sometimes new manufacturing models arise as such a situation leads to an increasing adoption of new production technologies. The challenge with distributed production is to implement communication and integration technologies that reduce the coordination effort and provide a focused factory (Khajavi et al. 2014).

In the proposed scenario, the Scattered Manufacturing Network is composed of AMT nodes that are able to manage instance orders or buy production resources (slots) coordinated by an orchestrator, responsible of the communication along the network, the negotiation among nodes and the overall (production and logistics) optimization along the supply chain structure. Further details about this topic, as well as numerical approaches, are in (De Falco et al. 2016) and (Tomasiello and Macìas – Dìaz, 2017).

The network obeys three principles: sustainability, shared resources and transparency. Sustainability occurs in terms of cost-effective manufacturing, reductions of resource demands and related CO_2 emissions over the entire product life cycle transferring the production closer to the client. According to Circular Economy trend, the SM Network aims to create a collaborative, transparent, open and trusting environment with shared purposes and shared resources. In fact, every node in the network can buy resources in the world with an open bidding system, while customers can send demands of products (orders) to the orchestrator. Hence, the orchestrator acts as an intermediate layer collecting orders from many customers.

In this paper, the authors present the just described paradigm via a SM Network within the 3D Printings (3DPs) field, where the different nodes are "productive" or "demanding". Productive nodes provide finished pieces realized by 3D printings. Demanding nodes, instead, orders finished pieces from the productive nodes. In this sense, demanding nodes formulate work orders that have to be satisfied by the productive ones. Considering complex dynamics inside large networks, it often happens that a same node can be of either productive or demanding type in different times and/or situations. This suggests that the communication and/or negotiation activities among nodes, managed by the orchestrator, are fundamental in order to share different resources and distribute them along the network by satisfying principles of transparency and sustainability. In what follows, we investigate the dynamics of networks with a unique demanding node and various productive ones. In general, such assumption is not detrimental for the discussion of a general issue but further details will be described in future research activities.

In order to consider the variables/factors that affect a 3DPs network, a unique model is proposed by combining different approaches, which focus on such needs:

- Logistics issues related to the productive nodes that are near the demanding one.
- Possibility of division of demanding node's order into subparts, and consequent assignment of each subpart to a productive node.
- Negotiation criteria between the demanding node and the productive ones, in order to establish tradeoffs between different profits.

Hence, the orchestrator has a primary role as it behaves like a control unit that applies a multilevel optimization that deals with the following exigencies. *Localization*: starting from the demanding node's geographical position, the orchestrator provides some neighboring nodes that define a "certified" sub-network, able to satisfy the users' requests. *Fragmentation* and *Assignment*: the orchestrator establishes how to divide the work order into subparts, each one assigned to different productive nodes, in order to achieve the lowest overall purchase cost. Notice that this phenomenon requires a suitable negotiation between the demanding node, which asks for a predefined amount of pieces, and the productive nodes, that have their own quantity/price plans. *Picking*: the orchestrator defines a closed path that starts from the demanding node and returns to it touching all the productive nodes once. Such path, useful to collect the amount of pieces from all the productive nodes, is obtained via an approach (see Karp, 1979; Gutin et al. 2002; Applegate et al. 2006; Gutin and Punnen, 2007; Gutin and Yeo, 2007; Wang et al. 2016) that minimizes the logistics costs.

Based on the just described requirements, the orchestrator set a run of iterations. As for the first one, the orchestrator:

- Indicates suitable productive nodes near the demanding one, and assigns them the amount of pieces to produce by satisfying constraints dealing with quantity/price plans.
- Defines a picking path at minimum logistics cost.
- Computes the weights that each productive node has inside the network. Precisely, for each node the corresponding weight represents a tradeoff among logistics components, possible quantities of produced pieces, as well as reallocation of quantities by excluding the productive node from the network in consideration. Such last operation is necessary to discriminate among different productive nodes that can be far from the demanding one (hence requiring high logistics costs) but in turn useful due to their advantageous quantity/price plans.

As for the second one, the orchestrator works as follows. First, the productive node, whose weight indicates the highest decrement of the overall logistics and production costs, is excluded from the network. Then, the orchestrator redefines either the picking path or new fragmentations/assignments to the remaining productive nodes. This last phenomenon triggers a consequent negotiation phase between the demanding node and the productive ones and the result is a tradeoff between different profits. Finally, the orchestrator recalculates the new weights of the remaining productive nodes, and next iterations works as the second one. Iterations continue until the computation of weights indicates that further decrements of costs are not possible, hence reaching an equilibrium state.

Indeed, the actual originality of the proposed approach foresees a complete balance among exigencies of different nodes. In fact, starting from localization requests of the demanding node that needs a certified service network, a unique framework mixes approaches for picking paths and resource allocation problems that solve issues of fragmentation and assignment. Such aspects are dependent each other, as they are strictly connected by the weights that the various productive nodes have inside the network. In fact, the possible exclusion of productive nodes from the network determines a guideline to solve at the same time logistics issues, as well as reallocations by considering the overall quantity/price plans of each productive node. This last aspect, which clearly deals with the negotiation phases between the demanding node and the productive ones, represents the effective dynamics of the network at each iteration provided by the orchestrator.

The presented numerical examples have either expected features or unexpected ones. For instance, it is possible that the exclusion of a node during the iterations could provoke increments of logistics costs, as well as suitable reduction of productive purchase. This implies the consequent birth of new iterations, where the tradeoff between logistics and production components indicates clearly that the nodes to exclude do not obey a predefined and precise rule. Moreover, it could occur that networks of medium dimensions reach an equilibrium state in just one iterations, and such a phenomenon is important as it indicates that larger networks are sometimes easier to manage.

The outline of the paper is as follows. Section II and III present the proposed approach and its features within the context of SM networks. Section IV contains simulation results for some test networks. Section V ends the paper with conclusions and future developments.

1* Dipartimento di Scienze Aziendali – Management & Innovation Systems,

University of Salerno, Via Giovanni Paolo II, 132, 84084, Fisciano (SA), Italy

Email: mdefalco@unisa.it

1* Dipartimento di Scienze Aziendali – Management & Innovation Systems,

University of Salerno, Via Giovanni Paolo II, 132, 84084, Fisciano (SA), Italy

Email: nmastrandrea@unisa.it

2* Dipartimento di Studi Aziendali e Quantitativi,

University Parthenope of Naples, Via Generale Parisi 13, Palazzo Pacanowski, 80132, Napoli

Email: lrarita@unisa.it

3* University of Dubai, United Arab Emirates

Email: ahalawin@ud.ac.ae

2. A 3DPS NETWORK

This section describes some features for a SM network within the context of 3DPs.

The SM network has the set of nodes $V = \{v_1, v_2, \ldots, v_N\}$ while:

- e_{ij} is the arc that connects nodes v_i and v_j;
- c_{ij} is the cost for arc e_{ij}.

Notice that c_{ij} depends on various factors, such as the distance between v_i and v_j, the monetary overall cost for transports, the traveling time, as well as criteria of sustainability. Parameters c_{ij} are kept in a coefficient matrix $X = \left(c_{ij}\right)_{i,j=1,\ldots,N}$.

The SM network is assumed to be bidirectional, namely: two different nodes v_i and v_j are connected in the direction either "from v_i to v_j" or "from v_j to v_i". Obviously, e_{ij} and e_{ji} are the same arc while, in general, $c_{ij} \neq c_{ji}$.

Each node provides services to the users in terms of finished pieces produced by 3DPs. Quantities Q_i and prices P_i of pieces for a generic node v_i obey a "law at three levels" of type:

$$P_i(Q_i) = \begin{cases} p_L^i & if\ 0 < Q_i \leq k_L^i \\ p_M^i & if\ k_L^i < Q_i \leq k_M^i \\ p_H^i & if\ k_M^i < Q_i \leq k_H^i \end{cases} \tag{1}$$

with $p_L^i < p_M^i < p_H^i$. The interpretation is the following: if the required quantity Q_i does not exceed k_L^i, the price P_i is the lowest p_L^i; otherwise, possible prices are p_M^i and p_H^i.

Notice that (1) represents a possible and realistic attempt to describe the evolution of pieces versus their possible prices. Indeed, future research activities aim at guaranteeing more suitable shapes for (1), with the aim of describing negotiation criteria among nodes.

3. OPTIMIZATION OF CUSTOMERS' NEEDS

This section shortly describes a possible approach for the optimization of demanding node's needs inside a 3DPs network. In particular, a unique framework is described in which more approaches, often used individually, are used.

The aim of the demanding node is to obtain a series of services from the SM network. In the specific case, in a preliminary phase, the orchestrator helps the demanding node referring to the following issues:

- *Localization*: the orchestrator makes the demanding node become the center of a circle with a radius of "economic" type. This means that, according to the demanding node's geographical position, only some production nodes, belonging to an area tracked by the orchestrator, are able to offer services. Such a localization criterion has the advantage of defining the most known and neighboring nodes, thus creating a sort of "certified" sub-network, for which the demanding node has a higher level of trust. Each available node of the certified sub-network shows its own offer in terms of prices/quantities plans.

- *Fragmentation and Assignment*: the orchestrator, considering the features of the sub-network, decides: how to fragment the demanding node's work order and how to assign the various subparts to the production nodes, in order to get the lowest purchase costs.
- *Picking*: the orchestrator chooses a closed path, which starts and returns to the demanding node through all the production nodes once. The path is defined via an approach described by procedure shown as follows (see for details Karp, 1979; Gutin et al. 2002; Applegate et al. 2006; Gutin and Punnen, 2007; Gutin and Yeo, 2007; Wang et al. 2016).

For a SM network whose features are described in previous section, the following procedure is used for the picking activities. Assume that P is a possible closed path that crosses each node of V, starting from a source node $v_s \in V$, and coming back to it; $C(P)$ is the cost associated to P.

Picking algorithm (PA).

Initialization:

$P := \emptyset, C(P) := 0, v_s := v_j \in V.$

Steps:

1. From node v_j go to node $v_i \in V \backslash \{v_j\}$ such that $c_{ji} = min \bigcup_{i=1,i \neq j}^{|V|} c_{ji}$.
2. $P \leftarrow P \cup e_{ji}, C(P) \leftarrow C(P) + c_{ji}, V \leftarrow V \backslash \{v_j\}$.
3. If $|V| = 1, P \leftarrow P \cup e_{is}$ *end of the algorithm*; otherwise, $v_j \leftarrow v_i$ and go to step 1.

For a better comprehension of such a procedure, see Example PA in the Appendix.

3.1 Combining issues of Localization, Fragmentation, Assignment and Picking

Considering a demanding node that asks for services from a generic SM network, the aim is to decrease the overall cost, that has two components, C_X and C_Y, that refer to the logistics (associated to the path) and the purchase costs, respectively.

In what follows, a suitable algorithm, that considers all exigencies of the demanding role, is presented.

Consider a preliminary phase (iteration 0). The orchestrator, referring to a SM network with N nodes (Figure 1, up left), tracks an economic radius for the demanding node (in position T in Figure 1, up right), discriminates the unreliable nodes (in red, Figure 1, bottom left), defines a closed path P from T to T according to algorithm (PA), and computes the weights of each production node inside the network.

Notice that P has production nodes for which, respecting constraints of fragmentation and assignment, purchase costs occur. Hence, at the iteration 0, C_X and C_Y are as follows: $C_X^{(0)} = C(P)$ and $C_Y^{(0)} = \sum_{i=1}^{N} p_L^i$.

FIGURE 1

Preliminary phase (iteration 0)
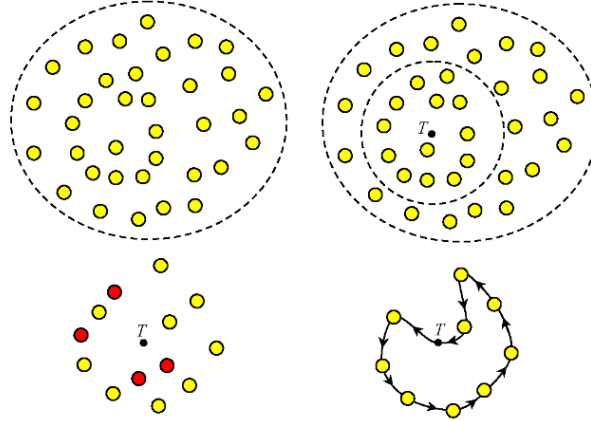
For further iterations, the orchestrator works as follows in order to decrease the overall costs:

- Erasing of nodes that can provoke the lowest costs in the next iteration.
- Reallocation of the sub-parts of the work orders (a new fragmentation and assignment phase), with consequent negotiation between the demanding node and the productive nodes. Notice that reallocation activities foresee a possible saturation of productive nodes.
- Computation of a new path for picking by using algorithm (PA).
- Computation of new weights (reallocation parameters) associated to the productive nodes.

Notice that erasing nodes from the network implies an obvious natural variation of either logistic or purchase costs. In order to understand the entity of variations and compute the weights for the productive nodes, we define the following quantities:

- $\Delta C_X^{v_i}$, that represents the variation of the logistics costs when node v_i is excluded from the network. Precisely, we have that $\Delta C_X^{v_i} = C(P) - C(P \backslash v_i)$.
- $\Delta C_Y^{v_i}$, that indicates the dynamics of purchase costs when node v_i is excluded from the network. In detail, we get that: $\Delta C_Y^{v_i} = -Q_i[P_i(Q_i)] + \sum_{j=1, j\neq i}^{|V|} Q'_j P_j (Q_j + Q'_j)$, where $P_i(Q_i)$ follows (1) while Q'_j is the amount of pieces, redistributed on the network, computed using algorithm (RA), described as follows.
- $\Delta R^{v_i} = \Delta C_X^{v_i} + \Delta C_Y^{v_i}$ is the weight (reallocation parameter) associated to node v_i. Notice that, if $\Delta R^{v_i} < 0$, the exclusion of node v_i allows a decrement of the overall cost for the network.

Reallocation algorithm (RA).

Assume that L is the total amount of pieces, which the demanding nodes requires from the network.

If node v_i is excluded from the network, Q_i is redistributed among nodes. The new quantity $Q'_j, j = 1, ..., |V|, j \neq i$, is defined as follows:

$$Q'_j = \left\lceil \frac{Q_i}{|V|-1} \right\rceil,$$

(2)

such that $\sum_{j=1,j\neq i}^{|V|} Q'_j = L$.

Notice that (2) has the following interpretation: as the productive nodes have all the same importance for the demanding node, the quantity Q_i is equally distributed among all the other remaining productive nodes. If $\frac{Q_i}{|V|-1}$ is not integer, then the full upper part is taken.

Finally, the overall optimization algorithm, defined by the orchestrator's activities, works as follows, at the n-th iteration:

- *Step 1*: Erase the node j_n whose weights allows a reduction of the overall costs for the network in consideration.
- *Step 2*: Compute a new path for picking, with new cost $C_X^{(n)}$.
- *Step 3*: Reallocate the quantities of pieces $Q_i, i = 1, i \neq j$ of each node according to algorithm (RA).
- *Step 4*: Compute the new weights for productive nodes.
- *Step 5*: Come back to step 1 if there is at least one reallocation parameter ΔR^{v_i} is negative.

Figure 2 provides an intuitive idea of the optimization algorithm, considering the second, the third and then the n-th iteration.

FIGURE 2

Left: in the second iteration, a node (in red) is excluded, the picking path is recomputed, the logistics cost decreases while the overall purchase one can remain the same or decrease. Center: in the third iteration, another node is excluded, and the process continues. Right: in the n-th iteration, the overall logistics and purchase costs are highly decreased
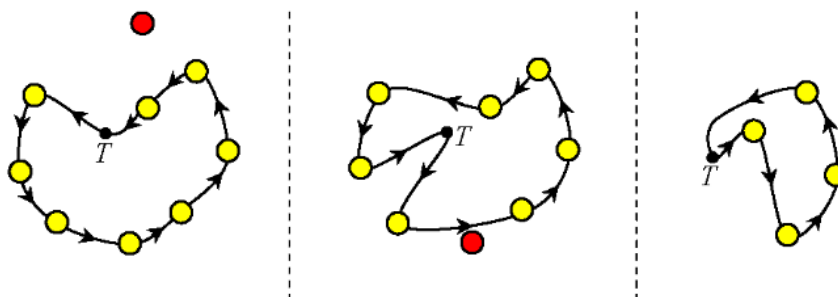
The following example shows the various step of the algorithm.

Example A.

Consider a SM network with $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ and matrix $X = \left(c_{ij}\right)_{i,j=1,\dots,6}$.

A possible interpretation of the starting phase (iteration 0) is in Figure 3. The preliminary closed path P (Figure 3, up) involves the demanding node, v_1, and the productive nodes v_2, v_3, v_4, v_5 and v_6. For the productive nodes, the orchestrator determines purchase costs of p_L type (Figure 3, down), together with suitable assignments of finished pieces, as well as weights of each productive node inside the network.

FIGURE 3

Example A, iteration 0
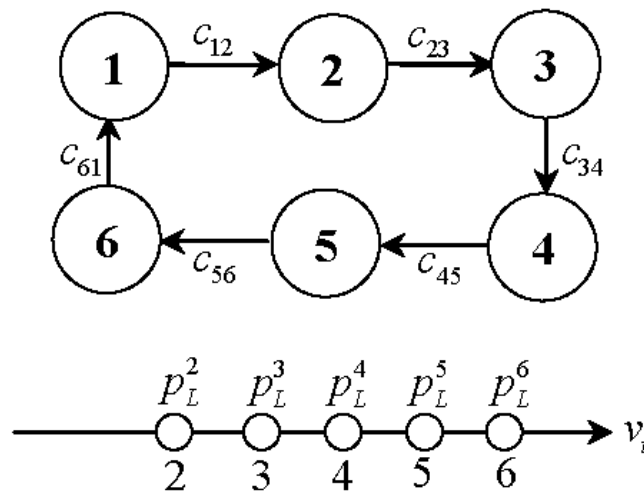
At the iteration 1:

- Node v_3 is excluded because it has the lowest reallocation parameter;
- The closed path P is recomputed using algorithm (PA) considering the set $V = \{v_1, v_2, v_4, v_5, v_6\}$;
- A cost $C_X^{(1)}$ is obtained;
- A new fragmentation/assignment is made for the nodes of set V, see algorithm (RA).
- A cost $C_Y^{(1)}$ is obtained.
- New weights for productive nodes are computed.

Figure 4 sums up the iteration 1.

FIGURE 4

Example A, iteration 1
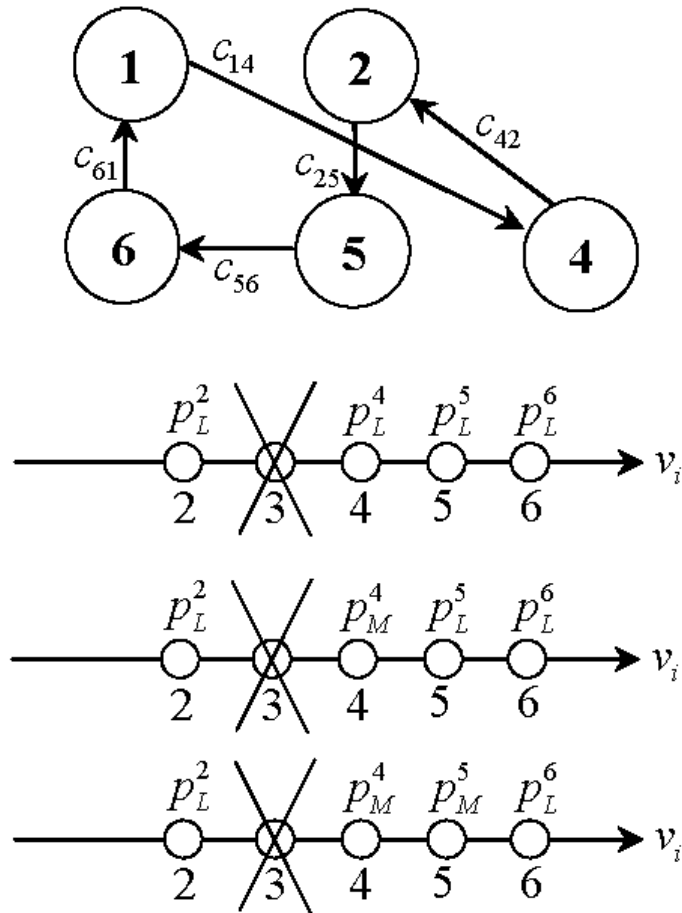
Precisely, Figure 4 (up) presents the new path, while Figure 4 (bottom) shows possible scenarios for $C_Y^{(1)}$:

- Scenario 1.1 (no variations): p_L^3 is erased and all other prices of p_L type remain the same.
- Scenario 1.2 (variations of just one cost): p_L^3 is erased, p_L^4 becomes p_M^4 and all other prices of p_L type remain the same.
- Scenario 1.3 (variations of more costs): p_L^3 is erased while, for instance, p_L^4 and p_L^5 become, respectively, p_M^4 and p_M^5.

Assuming that the scenario 1.2 occurs, at the iteration 2:

- Node v_2 is excluded due to its weight;
- The closed path P is obtained via algorithm (PA) for the new set $V = \{v_1, v_4, v_5, v_6\}$;
- A cost $C_X^{(2)}$ is computed;
- A new fragmentation/assignment occurs for the nodes of V, see algorithm (RA).
- A cost $C_Y^{(2)}$ is computed.
- New weights for productive nodes are established.

Figure 5 presents the iteration 2.

Figure 5

Example A, iteration 2
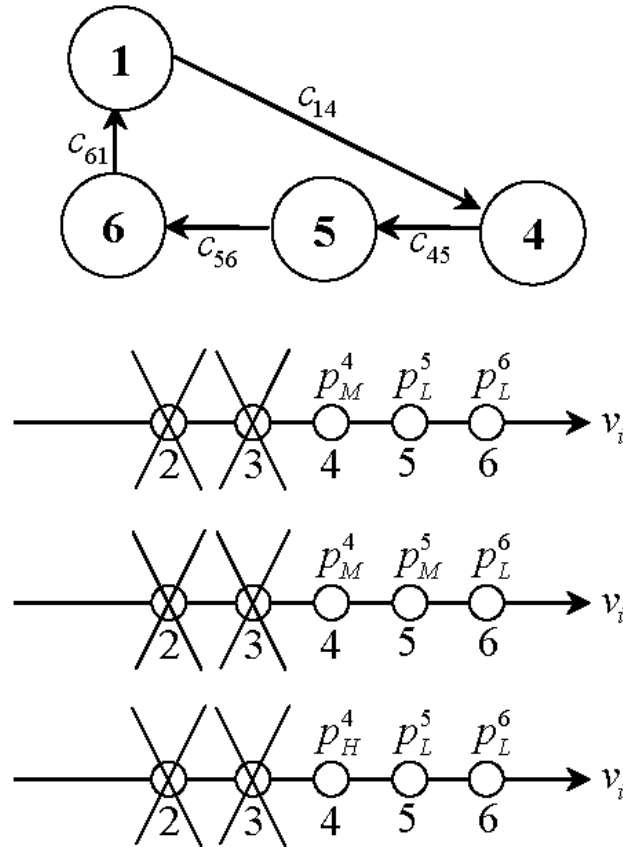
Precisely, Figure 5 (up) considers the new path, while Figure 5 (bottom) indicates possible new scenarios for $C_Y^{(2)}$:

- Scenario 2.1 (no variations): p_L^2 is erased while all other nodes have prices foreseen in scenario 1.2.
- Scenario 2.2 (variations of just one cost): p_L^2 is erased, p_L^5 becomes p_M^5 while all other nodes have prices described in the scenario 1.2.
- Scenario 2.3 (a high cost is achieved): p_L^2 is erased, p_M^4 becomes p_H^4 while all other nodes have prices described in scenario 1.2. In this case, the algorithm ends, as one term of p_H type is obtained, and all reallocation parameters become positive.

Assuming that the scenario 2.2 occurs, at the iteration 3:

- Node v_6 is excluded;
- A new closed path P is computed by algorithm (PA) for the new set $V = \{v_1, v_4, v_5\}$;
- A cost $C_X^{(3)}$ is obtained;
- A new fragmentation/assignment is made for the nodes of V, see algorithm (RA).
- A cost $C_Y^{(3)}$ is considered.
- Weights for productive nodes are updated.

Figure 6 presents the iteration 3.

Figure 6

Example A, iteration 3

In particular, Figure 6 (up) shows the new path, while Figure 6 (bottom) presents possible various scenarios for $C_Y^{(3)}$:

- Scenario 3.1 (no variations): p_L^6 is erased while all other nodes have prices foreseen in the scenario 2.2.
- Scenario 3.2 (variations of just one cost): p_L^6 is erased, p_M^4 becomes p_H^4 while all other nodes have prices described in scenario 2.2. The algorithm ends, as one term of p_H type is obtained and all reallocation parameters become positive.

4. NUMERICAL TESTS
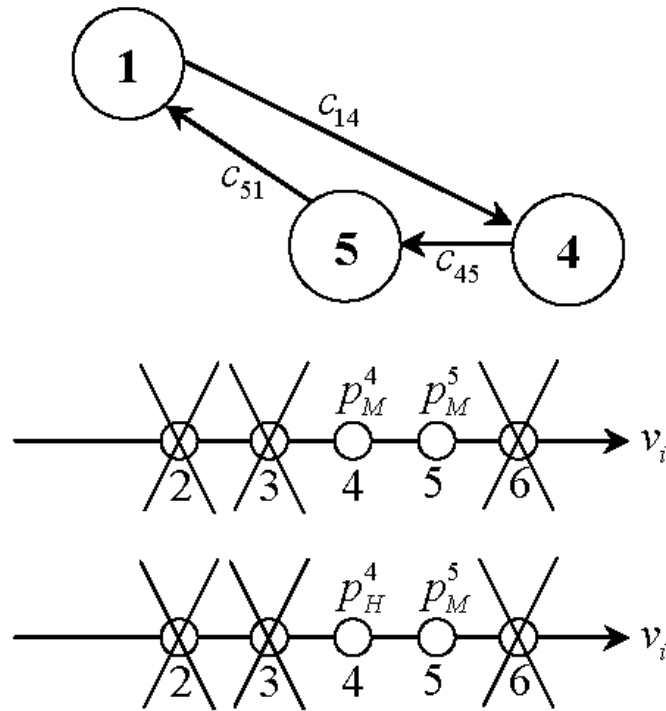
This section is devoted to some numerical tests. In particular, each of them presents some features that are useful to provide a better idea of dynamics inside a SM network.

Test 1.

Focus on a SM network with $V = \{v_1, v_2, v_3, v_4\}$ and matrix X:

$$X = \begin{pmatrix} \infty & 10 & 40 & 30 \\ 10 & \infty & 20 & 50 \\ 40 & 20 & \infty & 40 \\ 30 & 50 & 40 & \infty \end{pmatrix}.$$

Assume that $P := \emptyset$, $C(P) := 0$, $v_s := v_1$. Hence, the demanding node is v_1 while node v_j, $j = 2, 3, 4$, is of productive type. Price/quantity plans follow the formulation (1) with $p_L^2 = 25, p_L^3 = 20, p_L^4 = 15$; $p_M^2 = 30, p_M^3 = p_M^4 = 25$; $p_H^2 = 35, p_H^3 = p_H^4 = 45$; $k_L^2 = 40, k_L^3 = 50, k_L^4 = 70$; $k_M^2 = 50, k_M^3 = 90, k_M^4 = 95$; $k_H^2 = 60, k_M^3 = k_M^4 = 95$.

We need to distribute $L = 100$ pieces among the productive nodes v_2, v_3 and v_4. Preliminarily, the orchestrator indicates $Q_2 = 10$, $Q_3 = 30$ and $Q_4 = 50$. Such quantities are offered at prices p_L^2, p_L^3 and p_L^4, respectively.

The iterations run as follows.

First iteration:

From algorithm (PA), we have $P := \{e_{12}, e_{23}, e_{34}, e_{41}\}$, $C(P) := 100 = C_X^{(1)}$. As $Q_2^{(1)} = Q_2 = 10$, $Q_3^{(1)} = Q_3 = 30$ and $Q_4^{(1)} = Q_4 = 50$, from the preliminary fragmentation we get $C_Y^{(1)} = Q_2^{(1)} p_L^2 + Q_3^{(1)} p_L^3 + Q_4^{(1)} p_L^4 = 1600$. Therefore, $C_{TOT}^{(1)} = 1700$.

By considering the computation of weights for nodes, as well as the reallocation via algorithm (RA), Table 1 is obtained.

TABLE 1

Dynamics of logistics paths, costs and reallocations for the first iteration

Erase node v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_2	$\{e_{14}, e_{43}, e_{31}\}$	$+10$	-75	-65
v_3	$\{e_{12}, e_{24}, e_{41}\}$	-10	0	-10
v_4	$\{e_{12}, e_{23}, e_{31}\}$	-30	$+375$	$+345$

From Table 1, node 2 must be excluded in the next iteration.

Second iteration:

We get $P := \{e_{14}, e_{43}, e_{31}\}$, $C(P) := 110 = C_X^{(2)}$. Moreover, $Q_3^{(2)} = 35$ and $Q_4^{(2)} = 55$ at prices p_L^3 and p_L^4 and $C_Y^{(2)} = Q_3^{(2)} p_L^3 + Q_4^{(2)} p_L^4 = 1525$. Hence, $C_{TOT}^{(2)} = 1635$. Table 2E shows the possible reallocations and variations of costs for the next iteration.

TABLE 2

Reallocation for the second iteration

Erase node v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_3	$\{e_{14}, e_{41}\}$	-50	$+725$	$+675$
v_4	$\{e_{13}, e_{31}\}$	-30	$+275$	$+245$

As Table 2 indicates that $\Delta R^{v_i} > 0$, $i = 3, 4$, the possible exclusion of nodes v_3 and v_4 does not imply a reduction of the overall cost. Hence, the iterations stop.

From the following example, we consider two important phenomena. First, the exclusion of a node from the network does not necessarily imply a reduction of the logistics costs. This is due to the recalculation of the new picking path, which can be very different, also in terms of costs of arcs, from the ones of previous iterations. Second, for the last iteration node v_2 is not considered. At a first sight, one expects that this could occur for node v_4 as $\Delta C_X^{v_4} < \Delta C_X^{v_2}$. Indeed, as $\Delta C_Y^{v_4} \gg \Delta C_Y^{v_2}$, the possible exclusion of node v_4 implies the worst case for the overall cost, that increases considerably. Hence, although node v_2, unlike v_4, is the less advantageous in logistics terms for the demanding node v_1, it must avoided because of high fluctuations of production costs.

Test 2.

Consider a SM network with $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ and matrix X:

$$X = \begin{pmatrix} \infty & 10 & 25 & 5 & 7 & 9 & 13 \\ 14 & \infty & 171 & 21 & 12 & 5 & 12 \\ 11 & 15 & \infty & 14 & 13 & 12 & 14 \\ 11 & 10 & 9 & \infty & 17 & 12 & 13 \\ 15 & 12 & 11 & 9 & \infty & 14 & 18 \\ 12 & 13 & 17 & 17 & 19 & \infty & 15 \\ 13 & 11 & 9 & 13 & 15 & 17 & \infty \end{pmatrix}.$$

The demanding node is v_1 while node v_j, $j = 2, \dots 7$, is of productive type. For productive nodes, prince/quantity functions have levels shown in Table 3.

TABLE 3

Levels for prince/quantity plans of the network in consideration

Node i	p_L^i	p_M^i	p_H^i	k_L^i	k_M^i	k_H^i
v_2	25	35	45	15	25	35
v_3	35	45	55	20	40	50
v_4	30	40	60	25	35	50
v_5	20	30	45	10	25	40
v_6	30	45	60	15	25	35
v_7	30	40	50	25	40	55

Assume that $P \coloneqq \emptyset$, $C(P) \coloneqq 0$, $v_s \coloneqq v_1$. In this case, $L = 80$ pieces must be distributed among the productive nodes. At the beginning of the iterations, the orchestrator provides $Q_2 = 10$, $Q_3 = 15$, $Q_4 = 20$, $Q_5 = 5$, $Q_6 = 10$ and $Q_7 = 20$, at prices p_L^i, $i = 2, \dots, 7$.

The iterations run as follows.

First iteration:

From algorithm (PA), we have $P \coloneqq \{e_{14}, e_{43}, e_{36}, e_{62}, e_{25}, e_{57}, e_{71}\}$, $C(P) \coloneqq 82 = C_X^{(1)}$. As $Q_i^{(1)} = Q_i$, $i = 2, \dots, 7$, we simply have that $C_Y^{(1)} = \sum_{i=2}^{7} Q_i^{(1)} p_L^i = 2375$, and $C_{TOT}^{(1)} = 2457$.

As for the computation of weights for various nodes and reallocation due to algorithm (RA), Table 4 is obtained.

TABLE 4

Possible logistics paths, costs and reallocations for the first iteration

Erase v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_2	$\{e_{14}, e_{43}, e_{36}, e_{67}, e_{75}, e_{51}\}$	-11	$+40$	$+29$
v_3	$\{e_{14}, e_{42}, e_{26}, e_{67}, e_{75}, e_{51}\}$	-17	-120	-137
v_4	$\{e_{15}, e_{53}, e_{36}, e_{62}, e_{27}, e_{71}\}$	-14	$+100$	$+86$
v_5	$\{e_{14}, e_{43}, e_{36}, e_{62}, e_{27}, e_{71}\}$	-18	$+50$	$+32$
v_6	$\{e_{14}, e_{43}, e_{35}, e_{52}, e_{27}, e_{71}\}$	-18	$+50$	$+32$
v_7	$\{e_{14}, e_{43}, e_{36}, e_{62}, e_{25}, e_{51}\}$	-16	$+50$	$+34$

Table 4 foresees that node v_3 must not be considered in the second iteration.

Second iteration:

In this case, the new path is $P \coloneqq \{e_{14}, e_{42}, e_{26}, e_{67}, e_{75}, e_{51}\}$ and $C(P) \coloneqq 65 = C_X^{(2)}$. We get that $Q_2^{(1)} = 13, Q_4^{(1)} = 24, Q_5^{(1)} = 8, Q_6^{(1)} = 13, Q_7^{(1)} = 23$, while $C_Y^{(2)} = 2255$, and $C_{TOT}^{(1)} = 2320$.

As for the computation of reallocation parameters, we refer to Table 5.

TABLE 5

Variation of parameters for the second iteration

Erase node v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_2	$\{e_{14}, e_{46}, e_{67}, e_{75}, e_{51}\}$	-3	$+905$	$+902$
v_4	$\{e_{15}, e_{52}, e_{26}, e_{67}, e_{71}\}$	-13	$+800$	$+787$
v_5	$\{e_{14}, e_{42}, e_{26}, e_{67}, e_{71}\}$	-17	$+70$	$+53$
v_6	$\{e_{14}, e_{42}, e_{25}, e_{57}, e_{71}\}$	-7	$+925$	$+918$
v_7	$\{e_{14}, e_{42}, e_{26}, e_{65}, e_{51}\}$	-11	$+1010$	$+999$

The iterations stop because $\Delta R^{v_i} > 0$, $i = 2, 4, 5, 6, 7$. Moreover, the higher increments of terms $\Delta C_Y^{v_i}$, $i = 2, 4, 5, 6, 7$, are essentially due to the fact that prices become of type p_M. Such an event, indeed, does not always indicate very high discrepancies, as shown by $\Delta C_Y^{v_5}$.

Notice that the described example presents how a network of medium dimensions can reach an equilibrium situation in just one iteration. This suggests that suitable policies of choosing productive nodes could foresee to enlarge the economic radius, see Section 2, in order to achieve higher advantages in terms of lower costs.

Test 3.

We present a SM network with $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$ and matrix X:

$$X = \begin{pmatrix} \infty & 10 & 15 & 17 & 12 & 11 & 17 & 18 & 19 & 22 \\ 14 & \infty & 11 & 18 & 17 & 24 & 14 & 22 & 18 & 20 \\ 17 & 20 & \infty & 22 & 21 & 22 & 23 & 24 & 18 & 19 \\ 17 & 16 & 15 & \infty & 19 & 21 & 24 & 23 & 22 & 21 \\ 19 & 21 & 22 & 17 & \infty & 22 & 21 & 20 & 19 & 17 \\ 20 & 21 & 22 & 23 & 24 & \infty & 24 & 22 & 21 & 19 \\ 18 & 17 & 18 & 15 & 14 & 19 & \infty & 21 & 22 & 24 \\ 15 & 18 & 21 & 24 & 27 & 22 & 21 & \infty & 18 & 20 \\ 38 & 37 & 35 & 32 & 44 & 42 & 41 & 41 & \infty & 40 \\ 15 & 18 & 18 & 17 & 16 & 17 & 18 & 19 & 21 & \infty \end{pmatrix}.$$

We assume v_1 as the demanding node, while node v_j, $j = 2, \dots 7$, is productive. Levels of prince/quantity plans are in Table 6.

TABLE 6

Prince/quantity plans for the network in consideration

Node i	p_L^i	p_M^i	p_H^i	k_L^i	k_M^i	k_H^i
v_2	10	12	15	20	25	35
v_3	15	18	20	25	30	40
v_4	10	11	12	25	35	50
v_5	15	17	19	35	45	50
v_6	25	28	30	30	35	55
v_7	20	22	24	25	30	45
v_8	15	18	21	15	20	25
v_9	10	12	14	20	30	40
v_{10}	10	13	15	25	30	40

Preliminarily, $P := \emptyset$, $C(P) := 0$, $v_s := v_1$. We consider $L = 130$ pieces, which have to be distributed among the nine productive nodes. When the iterations start, the orchestrator indicates the following division: $Q_2 = 10$, $Q_3 = 15$, $Q_4 = 15$, $Q_5 = 25$, $Q_6 = 20$, $Q_7 = 15$, $Q_8 = 5$, $Q_9 = 10$ and $Q_{10} = 15$, at prices p_L^i, $i = 2, \dots, 9$.

The iterations are listed as follows.

First iteration:

From algorithm (PA), we have $P := \{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{106}, e_{68}, e_{87}, e_{71}\}$, $C(P) := 185 = C_X^{(1)}$. We have that $Q_i^{(1)} = Q_i$, $i = 2, \ldots, 9$, hence $C_Y^{(1)} = \sum_{i=2}^{9} Q_i^{(1)} p_L^i = 1975$, and $C_{TOT}^{(1)} = 2160$.

Considering the various reallocations and variations of costs, we get Table 7.

TABLE 7

Logistics paths, variations of costs and reallocations for the first iteration

Erase node v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_2	$\{e_{16}, e_{610}, e_{105}, e_{54}, e_{43}, e_{39}, e_{97}, e_{78}, e_{81}\}$	-12	$+40$	$+28$
v_3	$\{e_{12}, e_{27}, e_{75}, e_{54}, e_{46}, e_{610}, e_{108}, e_{89}, e_{91}\}$	-15	-20	-35
v_4	$\{e_{12}, e_{23}, e_{39}, e_{910}, e_{105}, e_{58}, e_{87}, e_{76}, e_{61}\}$	-10	$+65$	$+55$
v_5	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{46}, e_{610}, e_{107}, e_{78}, e_{81}\}$	-20	-30	-50
v_6	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{107}, e_{78}, e_{81}\}$	-24	-245	-269
v_7	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{106}, e_{68}, e_{81}\}$	-24	-95	-119
v_8	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{106}, e_{67}, e_{71}\}$	-19	-20	-39
v_9	$\{e_{12}, e_{23}, e_{310}, e_{105}, e_{54}, e_{46}, e_{68}, e_{87}, e_{71}\}$	-30	$+40$	$+10$
v_{10}	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{58}, e_{87}, e_{76}, e_{61}\}$	-15	$+130$	$+115$

Table 7 shows that node v_6 has to be excluded in the second iteration.

Second iteration:

The new path becomes $P := \{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{107}, e_{78}, e_{81}\}$ and $C(P) := 161 = C_X^{(2)}$. We get that $Q_2^{(2)} = 13, Q_3^{(2)} = 17, Q_4^{(2)} = 17, Q_5^{(2)} = 28, Q_7^{(2)} = 17,$ $Q_8^{(2)} = 7, Q_9^{(2)} = 13$ and $Q_{10}^{(2)} = 18$, while $C_Y^{(2)} = 1730$, and $C_{TOT}^{(2)} = 1891$.

The computation of weights for nodes and possible variations of costs are presented in Table 8.

TABLE 8

Logistics paths, variations of costs and reallocations for the second iteration

Erase node v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_2	$\{e_{15}, e_{54}, e_{43}, e_{39}, e_{910}, e_{107}, e_{78}, e_{81}\}$	-5	$+40$	$+35$
v_3	$\{e_{12}, e_{27}, e_{75}, e_{54}, e_{410}, e_{108}, e_{89}, e_{91}\}$	-10	-45	-55
v_4	$\{e_{12}, e_{23}, e_{39}, e_{910}, e_{105}, e_{58}, e_{87}, e_{71}\}$	-7	$+55$	$+48$
v_5	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{410}, e_{107}, e_{78}, e_{81}\}$	-15	-60	-75
v_7	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{108}, e_{81}\}$	-20	-140	-160
v_8	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{107}, e_{71}\}$	-18	$+85$	$+67$
v_9	$\{e_{12}, e_{23}, e_{310}, e_{105}, e_{54}, e_{48}, e_{87}, e_{71}\}$	-26	$+40$	$+14$
v_{10}	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{58}, e_{87}, e_{71}\}$	-12	$+70$	$+58$

From Table 8, it follows that the next iteration does not foresee node v_7.

Third iteration:

In this case, the new path is $P := \{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{108}, e_{81}\}$ and $C(P) := 141 = C_X^{(3)}$. We get that $Q_2^{(3)} = 15, Q_3^{(3)} = 19, Q_4^{(3)} = 20, Q_5^{(3)} = 30,$ $Q_8^{(3)} = 9, Q_9^{(3)} = 16$ and $Q_{10}^{(3)} = 20$, while $C_Y^{(3)} = 1590$, and $C_{TOT}^{(3)} = 1731$.

Weights for nodes and variations of costs are in Table 9.

TABLE 9

Logistics paths, variations of costs and reallocations for the third iteration

Erase node v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_2	$\{e_{15}, e_{54}, e_{43}, e_{39}, e_{910}, e_{108}, e_{81}\}$	-5	$+35$	$+30$
v_3	$\{e_{12}, e_{25}, e_{54}, e_{410}, e_{108}, e_{89}, e_{91}\}$	-1	-65	-66
v_4	$\{e_{12}, e_{23}, e_{39}, e_{910}, e_{105}, e_{58}, e_{81}\}$	-11	$+45$	$+34$
v_5	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{410}, e_{108}, e_{81}\}$	-15	-16	-31

v_8	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{510}, e_{101}\}$	-19	-25	-44
v_9	$\{e_{12}, e_{23}, e_{310}, e_{105}, e_{54}, e_{48}, e_{81}\}$	-30	$+35$	$+5$
v_{10}	$\{e_{12}, e_{23}, e_{39}, e_{94}, e_{45}, e_{58}, e_{81}\}$	-16	$+45$	$+29$

Table 9 shows that node v_3 must be excluded in the next iteration.

Fourth iteration:

In this case, the new path is $P := \{e_{12}, e_{25}, e_{54}, e_{410}, e_{108}, e_{89}, e_{91}\}$ and $C(P) := 140 = C_X^{(4)}$. We get that $Q_2^{(4)} = 19, Q_4^{(4)} = 24, Q_5^{(4)} = 33, Q_8^{(4)} = 12, Q_9^{(4)} = 19$ and $Q_{10}^{(4)} = 23$, while $C_Y^{(4)} = 1525$, and $C_{TOT}^{(4)} = 1665$.

Weights for nodes and variations of costs are in Table 10.

TABLE 10

Logistics paths, variations of costs and reallocations for the fourth iteration

Erase node v_i	$P \backslash v_i$	$\Delta C_X^{v_i}$	$\Delta C_Y^{v_i}$	ΔR^{v_i}
v_2	$\{e_{15}, e_{54}, e_{410}, e_{108}, e_{89}, e_{91}\}$	-15	$+315$	$+300$
v_4	$\{e_{12}, e_{25}, e_{510}, e_{108}, e_{89}, e_{91}\}$	-21	$+349$	$+328$
v_5	$\{e_{12}, e_{24}, e_{410}, e_{108}, e_{89}, e_{91}\}$	-16	$+299$	$+283$
v_8	$\{e_{12}, e_{25}, e_{54}, e_{410}, e_{109}, e_{91}\}$	-16	$+139$	$+123$
v_9	$\{e_{12}, e_{25}, e_{54}, e_{410}, e_{108}, e_{81}\}$	-41	$+240$	$+199$
v_{10}	$\{e_{12}, e_{25}, e_{54}, e_{49}, e_{98}, e_{81}\}$	-125	$+287$	$+162$

The iterations stop as $\Delta R^{v_i} > 0$, $i = 2, 3, 4, 5, 8, 9, 10$. The network has reached an equilibrium. Notice that further iterations could be possible if the demanding nodes and production nodes could negotiate about the price/plans. This is object of further research activities, dealing with possible variations of parameters k_L, k_M and k_H.

5. CONCLUSIONS AND FUTURE DEVELOPMENTS

In this paper, the authors have shown a Scattered Manufacturing Network within the 3D Printings context. An orchestrator describes communication and dynamics along the network in order to establish tradeoffs between the exigencies of demanding and productive nodes.

In particular, a unique framework has been used to describe logistics and reallocation issues, while suitable numerical examples have tested the proposed approach.

The just described context paves the way for further research activities. In the future, the authors aim to discuss complex manufacturing strategies for productive nodes, with consequent modelling of richer phenomena inside negotiation phases. Such phases will be defined by Multi-agent Systems. For them each node will be considered as smart agents that negotiate among themselves in order to pursue economic interests and manufacturing optimization needs (Leitão 2009; Preux et al. 2004; Wang et al. 2016) in a Peer-to-Peer decentralized architecture. This will also allow determining new logistics different approaches based on big data analysis, as well as introducing new scheduling issues dealing with temporal variables.

References

Anderl, R 2015, Industrie 4.0—technological approaches, use cases, and implementation. *Automatisierungstechnik*, pp. 753–765.

Applegate DL, Bixby RE, Chvátal, V, and Cook WJ 2006, The Traveling Salesman Problem: A Computational Study, Princeton University Press.

Ariss, S, Raghunathan, TS, and Kunnathar, A 2000, Factors affecting the adoption of advanced manufacturing technology in small firms. *S.A.M. Adv. Manag. J*, Vol. 65, No. 2.

Brettel, M, Friederichsen, N, Keller, and Rosenberg, M 2014, How virtualization, decentralization and network building change the manufacturing landscape: an Industry 4.0 perspective, *Int J Mech Aerospace. Ind. Mechatronics*, Vol. 8, No. 1, pp. 37–44.

Chen, F, Deng, P, Wan, J, Zhang, D, Vasilakos, AV and Rong, X 2015, Data mining for the internet of things: literature review and challenges. International Journal of Distributed Sensor Networks, Vol. 2015, Article ID 431047, 14 pages.

De Falco, M, Gaeta, M, Loia, V, Rarità, L, and Tomasiello S 2016, Differential quadrature-based numerical solutions of a fluid dynamic model for supply chains. *Commun. Math. Sci.*, Vol. 14, No. 5, pp. 1467–1476.

Durão, LFCS, Christ, A, Zancul, E, Anderl, R, and Schützer, K 2017, Additive manufacturing scenarios for distributed production of spare parts, *Int. J. Adv. Manuf. Technol.*, pp. 1-12.

Gutin, G. and Punnen AP (eds.) 2007, The Traveling Salesman Problem and its Variations, Kluwer, 2002 and Springer-Verlag.

Gutin, G, Yeo, A, and Zverovitch, A 2002, Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP, *Discrete Appl. Math.*, Vol. 117, pp. 81–86.

Gutin, G, and Yeo, A 2007, The Greedy Algorithm for the Symmetric TSP. *Algorithmic Oper. Res.*, Vol. 2, pp. 33–36.

Hauder, VA, Beham, A, Wagner, S and Affenzeller, M 2017 'Optimization networks for real-world production and logistics problems', Proc. Genet. Evol. Comput. Conf. Companion - GECCO '17, pp. 1411–1414.

Jonsson, P 2000, An empirical taxonomy of advanced manufacturing technology. *Int. J. Oper. Prod. Manag*, Vol. 20, No. 12, pp. 1446 –1474.

Karp, RM 1979, A patching algorithm for the non-symmetric traveling salesman problem, *SIAM J. Comput.*, Vol. 8, No. 4, pp. 561–573, 1979.

Khajavi, SH, Partanen, J, and Holmström, J 2014, Additive manufacturing in the spare parts supply chain. *Comput. Ind.*, Vol. 65, No. 1, pp. 50–63.

Leitão, P 2009, Agent-based distributed manufacturing control: a state-of the-art survey. *Eng. Appl. Artif. Intell.*, Vol. 22, No. 7, pp. 979–991.

MacDougall, W 2014. Industrie 4.0: Smart Manufacturing for the Future. Germany Trade & Invest.

Oliff H, and Liu, Y 2017, Towards Industry 4.0 Utilizing Data-Mining Techniques: A Case Study on Quality Improvement, *Procedia CIRP*, Vol. 63, pp. 167–172.

Preux, P, Delepoulle, S, and Darcheville, JC 2004, A generic architecture for adaptive agents based on reinforcement learning. *Inform. Sci.*, Vol. 161, No. 1, pp. 37–55.

Scheuermann, C, Verclas, S, and Bruegge, B 2015, Agile Factory-An Example of an Industry 4.0 Manufacturing Process, Proc. - 3rd IEEE Int. Conf. Cyber-Physical Syst. Networks, Appl. CPSNA 2015, pp. 43–47.

Schroder, R, Sohal, AS 1999, Organizational characteristics associated with AMT adoption: towards a contingency framework. *Int. J. Oper. Prod. Manag*, Vol. 19, No. 12, pp. 1270 –1291.

Tomasiello, S, Macías-Díaz, JE 2017, Note on a picard-like method for caputo fuzzy fractional differential equations. Appl. Math. Inform. Sci., Vol. 11, No.1, pp. 281–287.

Wan, J, Minglun, Y, Li, D, Zhang, C, Wang, S, and Zhou, K 2016, Mobile services for customization manufacturing systems: an example of industry 4.0, *IEEE Access*, Vol. 4, pp. 8977–8986.

Wang, S, Wan, J, Zhang, D, Li, D, and Zhang, C 2016, Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination. *Comput. Netw.*, Vol. 101, pp. 158–168.

Weiser, M 1993, Some computer science issues in ubiquitous computing, Communications of the ACM - Special issue on computer augmented environments: back to the real world, Vol. 36, No. 7, pp. 75–84.

Appendix

Example PA.

Consider a SM network with $V = \{v_1, v_2, v_3, v_4\}$ and matrix X:

$$X = \begin{pmatrix} 0 & 5 & 9 & 1 \\ 3 & 0 & 7 & 11 \\ 7 & 9 & 0 & 8 \\ 2 & 12 & 6 & 0 \end{pmatrix}.$$

Preliminarily, $P := \emptyset$, $C(P) := 0$, $v_s := v_1$. Table 11 shows the various iterations. Figure 1A, where numbers indicates the nodes for simplicity, presents a graphical evolution of the path.

Table 11

Evolution of the iterations

Iteration	P	$C(P)$	V
1	$\{e_{14}\}$	1	$\{2,3,4\}$
2	$\{e_{14}, e_{43}\}$	7	$\{2,3\}$
3	$\{e_{14}, e_{43}, e_{32}\}$	16	$\{2\}$
4	$\{e_{14}, e_{43}, e_{32}, e_{21}\}$	19	$\{2\}$

Figure 7

Iteration 1 (up, left): the first arc of the path P connects nodes v_1 and v_4. Iteration 2 (up, right): the path P connects nodes v_4 and v_3. Iteration 3 (bottom, left): the path P connects nodes v_3 and v_2. Iteration 4 (bottom, right): the path P connects nodes v_3 and v_2