

Optimal configurations of Minimally Intelligent additive manufacturing machines for Makerspace production environments

James Gopsill , Mark Goudswaard, Chris Snider , Lorenzo Giunta and Ben Hicks

Faculty of Science and Engineering, University of Bristol, UK

Research Article

Cite this article: Gopsill J, Goudswaard M, Snider C, Giunta L and Hicks B (2024). Optimal configurations of Minimally Intelligent additive manufacturing machines for Makerspace production environments. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, **38**, e2, 1–15
<https://doi.org/10.1017/S0890060423000239>

Received: 22 June 2023

Revised: 09 October 2023

Accepted: 25 November 2023

Keywords:

Makerspaces; Hackspaces; Hackerspaces; job-shops; workshops; FabLabs; additive manufacturing; material extrusion; agent-based modelings; minimal intelligence

Corresponding author:

James Gopsill;

Email: james.gopsill@bristol.ac.uk

Abstract

Additive manufacturing (AM) has transformed job shop production and catalysed the growth of Makerspaces, FabLabs, Hackspaces, and Repair Cafés. AM has enabled the handling and manufacturing of a wide variety of components, and its accessibility has enabled more individuals to make. While smaller than their production-scale counterparts, the objectives of minimizing technician overhead, capital expenditure, and job response time remain the same. The typical First-Come First-Serve (FCFS) operating model, while functional, is not necessarily the most efficient and makes responding to a-typical or urgent demand profiles difficult. This article reports a study that investigated how AM machines configured with Minimally Intelligent agents can support production in these environments. An agent-based model that simulated 5, 10, 15, and 20 AM machines operating a 9 am–5 pm pattern and experiencing a diverse non-repeating demand profile was developed. Machines were configured with minimal intelligence – FCFS, First-Response First-Serve (FRFS), Longest Print Time (LPT), Shortest Print Time (SPT), and Random Selection logics – that governed the selection of jobs from the job pool. A full factorial simulation totaling 15,629 configurations was run until convergence to a ranked list of production performance – min Job Time-in-System. Performance changed as much as 200%. Performant configurations featured a variety of logics, while the least performant were dominated by FCFS and LPT. All FCFS (a proxy for today’s operations) was one of the least performant configurations. The results provide an optimal set of logics and performance bands that can be used to justify capital expenditure and AM operations in Makerspaces.

Introduction

Affordable, capable, and flexible desktop material extrusion (MEX) and vat photopolymerization additive manufacturing (AM) technologies have transformed workshops and job-shops, enabling them to rapidly respond to a diverse range of jobs. Workshops in an education setting provide students with practical learning of manufacturing processes and prototype components, and the ability to create products for their projects. Workshops in industry provide essential services to prototype and test designs.

AM has also been a catalyst and, in some cases, revived the cottage industry with the introduction of Makerspaces, FabLabs, Hackspaces, Re-Makerspaces, and Repair Cafés across society (Scalfani and Sahib, 2013; Pryor, 2014; Nagle, 2021) (Fig. 1). Estimates suggest that there are over 6,000 “Makerspaces” across the globe (NESTA, 2015; Anon, 2023a, 2023b). These businesses provide societal value through the following:

- Education – by teaching society the art of “making”;
- Sustainability and the Circular Economy – by supporting the production of spare parts and remanufacturing to maintain goods’ active use; and,
- Innovation – by supporting start-ups and entrepreneurs in creating products for society’s consumption.

The reasons for AM’s proliferation and success in these environments include the democratization of the manufacturing process through a fully digitized Design & Manufacture (D&M) pipeline; the ability to support a wide variety of designs and material combinations; and the ability to be deployed in a wide variety of environments and locations (Ford and Despeisse, 2016; Smith and Mortati, 2017; Rautray and Eisenbart, 2021). Their small footprint and capital investment have also enabled these environments to deploy multiple machines, providing them with a step-change in production capacity.

It is therefore common for Makerspace environments to feature multiple AM machines that need to be supported by the respective technical teams. The technical teams require deep-domain skills to maintain and operate the machines as well as a wealth of knowledge on the design of components that can be manufactured using the machines (Annear et al., 2023). While smaller than their production-scale counterparts, the objectives of

© The Author(s), 2024. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/4.0>), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.



Figure 1. Examples of Makerspace environments.

- minimizing technician overhead in managing job workflows so technicians can spend more time deploying and sharing their skills and knowledge;
- minimizing capital expenditure by achieving more with the manufacturing capability; and,
- minimizing job response time to satisfy the clients of the service remain the same.

Job management has typically followed First-Come First-Serve (FCFS) principles (Gopsill and Hicks, 2018). Jobs are either queued up based on submission or individuals coming into the space to identify and check the availability of the machine(s) they wish to use. The methods are easy to implement and reliable.

However, FCFS has also proven problematic with the increasing variety and volume of jobs being received. This has resulted in low user satisfaction and productivity, extended development lead times, and difficulty in managing a-typical demand profiles or demand spikes.

A concept that fits the job management requirements of Makerspaces is agent-based manufacturing and, in particular, Minimally Intelligent agents (Gopsill *et al.*, 2022). As the name suggests, Minimally Intelligent agents feature the minimal intelligence required to assess their own parameters, negotiate with other agents, and make decisions to satisfy their goals (Cliff, 1997; Lomas and Cliff, 2020). In the case of manufacturing, jobs and machines are represented by agents. A Job agent's goal is to have its component(s) manufactured and a machine agent's goal is to have its machines manufacturing. Agents enter networks where they can negotiate to resolve their goals.

Minimally intelligent agents require little configuration and maintenance, making them ideally suited to Makerspaces (Giunta *et al.*, 2022, 2023). Their computational requirements are also minimal, enabling them to be placed on computationally constrained resources, such as the spare computational resource available on AM machine microcontrollers (Chung and Cheol-HeeYoo, 2013; Purusothaman *et al.*, 2013; Pantoja *et al.*, 2018). An unanswered research question lies in understanding how the configuration of agents across a set of AM machines experiencing an

a-typical Makerspace demand profile affects the performance of the Makerspace to deliver components to its customers.

This article answers this question by modeling the responsiveness of Minimally Intelligent agents experiencing a diverse Makerspace demand profile. The configurations were ranked according to the sum Job Time-in-Pool (TiP) scores. Correlations between number of messages and rejections, machine utilization, job characteristics, and system response were also examined.

The article continues with a related work section that reviews Makerspace and Makerspace-like environment practices (herein, referred to as Makerspace environments), the demand profiles they experience, and research into Minimally Intelligent agent manufacturing systems (section "Related work"). The article then describes the model used to investigate the responsiveness of alternate Minimally Intelligent agent configurations for diverse demand (section "Numerically modeling a Minimally Intelligent Agent-Based Makerspace"). The details of the full factorial study are then reported where the model was used to simulate Makerspace environments featuring 5, 10, 15, and 20 machines, operating 9 am–5 pm, and configured with one of five logics (section "Examining Minimally Intelligent Makerspace manufacturing performance"). The data was used to rank the Minimally Intelligent configurations, analyze the nature of the ranking, and assess responsiveness from a system, machine, and job perspective (section "Results"). This is followed by a discussion of the significance of the findings and how they can be used to support Makerspace environment setup, configuration, and operation (section "Discussion"). The article then concludes with a summary of the scientific contributions (section "Conclusion").

Related work

To situate the work, this section provides a summary of Makerspace environment operations, including the scheduling methods deployed and the increased application of AM. The section then summarizes research into Minimally Intelligent agent manufacturing systems.

Makerspace environments

Makerspace environments are used for a wide variety of making applications, ranging from Student Projects and Hackathons to manufacturing Personal Protection Equipment during the coronavirus pandemic (Daoulas et al., 2021; Longhitano et al., 2021; Goudswaard et al., 2022). The diverse use cases naturally manifest diverse demand profiles that exhibit a large variance in the type, quantity, and arrival time of jobs that need making. This is corroborated by Wilczynski’s (2015) review of Makerspaces in Engineering Design that highlighted the diverse composition of manufacturing capability, job requirements, and submission profiles environments. Wilczynski (2015) also highlighted there is rarely any consistency or repeatability in the submission profiles. This makes effective management of job flows a challenge and requires a Makerspace to be responsive.

Heragu et al. (2002) discussed how existing job-shop scheduling methodologies that “type” jobs via means of hierarchical structures can be quickly “undone” by the diversity of jobs submitted to Makerspaces (Lu et al., 2014). This is because the means of classifying the diverse set of jobs is non-trivial and often shifts the problem to developing and testing various classifications with alternate job-shop scheduling methods to find an optimum configuration. There is also no means to handle jobs that have “yet to be observed,” and determine how they should be classified and whether continuous re-classification is required for a scheduling method to maintain optimal operation.

Basuki et al. (2020) recognized the often unique composition of the low technician to machine/manufacturing process ratio across Makerspaces. They noted that technicians spend much of their time maintaining the capability rather than sharing their knowledge and manufacturing know-how with users of the service. In addition, there is little capacity and funds available to maintain and manage platforms aimed at managing job workflows (Schonwetter and Van Wiele, 2018; Mersand, 2021).

Job scheduling in Makerspaces has often operated an FCFS model or a variation of it. The rationale is that it offers an easy-to-understand interface which increases the likelihood of uptake and adoption by the community. Simplicity is a further reason why formalized production-based process workflows have gained little traction. However, it does prove problematic, with Makerspaces being unable to cope with the often chaotic and diverse demand. The result is delay that leaves users frustrated.

Interest in this problem has grown with recent research looking at optimizations in AM machine–job workflows (Oh et al., 2020). One of the first challenges in the workflow of jobs through AM

machines was the identification of failed prints in order to eliminate material waste and lost print time. This has been tackled through a few different approaches. Computer vision has been used to evaluate layer adhesion can be monitored and identify “spaghetti” 3D prints (Baumann and Roller, 2016; Paraskevoudis et al., 2020; Petsiuk and Pearce, 2020). And, filament spool sensors have been deployed to monitor for jams in filament extrusion (Aidala et al., 2022).

With the increasing confidence that prints will succeed first-time, Gopsill and Hicks (2016, and 2018) turned to the development of methods to optimize print operations. They started with the extraction of individual part G-Code from multi-part G-Code files in a job pool (Fig. 2). The parts were then re-positioned to optimize the utilization of AM bed space. This minimized change-over times and technician/user interaction, with the system resulting in increased productivity with machines manufacturing for longer portions of the day.

The industry has also been creating solutions for managed AM, see for example the Ultimaker Digital Factory, that provide facilities with top-down management for AM machines (Ultimaker, 2023). However, many interface only with a specific set of machines, which leads to challenges when Makerspaces offer capability from multiple AM suppliers.

In summary, the related work in Makerspaces highlights that the environments provide a unique job-scheduling problem due to the diversity in demand and manufacturing capability. The introduction of AM has provided a means to handle this diversity, although optimal operation of AM machines in these environments has yet to be developed and adopted. Table 1 summarizes the features and the consequence they have on job scheduling that makes the problem interesting, unique, and non-trivial to solve using existing adopted methods.

Minimally intelligent agent-based manufacturing systems

Minimally Intelligent Agent-Based Manufacturing Systems introduces the concept of manufacturing machines and jobs that individually reason and decide their own strategies for processing work through the system (Priore et al., 2001, 2014). This is diametrically opposed to the dominant industry practice of centralized manufacturing system control and governance (Eyers, 2018; Chen, 2019). Further, Minimally Intelligent agents refer to the ability to embed the capability on machine microcontrollers, which are resource-constrained and thereby unlikely to be able to deploy large Artificial Intelligent (AI) models, such as Deep Learning Neural Networks, in

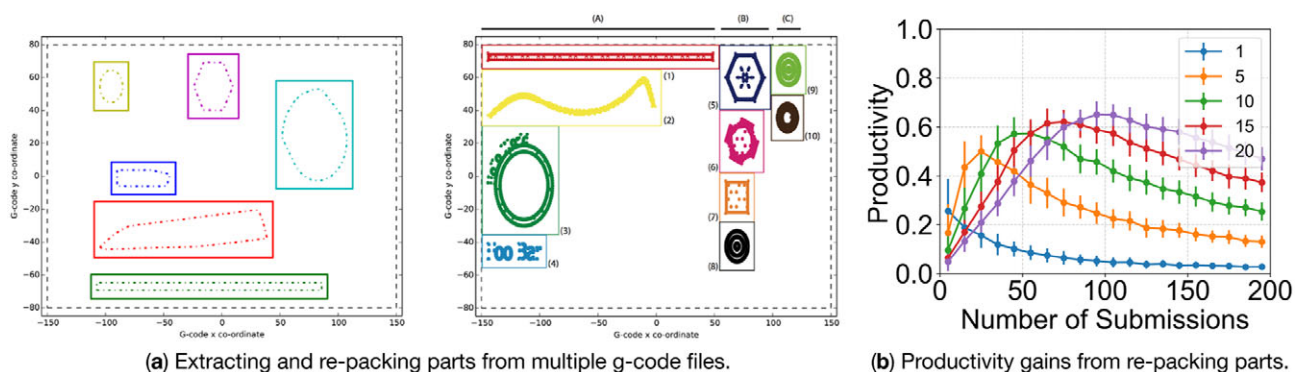


Figure 2. Re-packing parts from multiple G-Code submissions (from: Gopsill and Hicks (2018)).

Table 1. Features that make Makerspaces a unique and challenging job-scheduling problem

#	Feature	Consequence on scheduling
1	Diverse demand profiles	Difficult to impossible to forecast and plan schedules of work ahead of time. A scheduling method would have to react in real-time
2	Non-repeating demand profiles	
3	Low technician support to manufacturing capability ratio	Scheduling methods should be automated and require as little human intervention as possible
4	Minimal spare resource to support production-oriented scheduling tools	Scheduling algorithms need to be lightweight and be able to process on the existing available resource tools
5	AM machines supplied from multiple suppliers	Scheduling methods need to be made open and be able to interface with all manner of AM machine

the near term. The vision therefore attempts to make the most out of the compute resource available rather than require additional resource to operate (e.g., cloud high-performance computing¹).

Ma et al.'s (2021) numerical model of a Minimally Intelligent Agent-Based manufacturing system showed it to be more robust, resilient, and responsive compared to centralized control. Goudswaard et al. (2021) focused on sudden changes in demand behavior with a numerical model of a Minimally Intelligent Agent-Based manufacturing system configured to handle a steady-state demand input that then experienced a step, ramp, or saw-tooth change (Fig. 3a). The study revealed that different configurations were required to respond effectively to different changes in demand.

Obi et al. (2022) demonstrated how minimal intelligence logics that switched job priority based on the composition of the incoming demand were able to respond to sudden changes in job submissions (Fig. 3b). Following the demand change, the system would slowly return to its original state, although some configurations resulted in more machines remaining on one type of job even though there was a steady-state stream of jobs equally distributed across the job types. The study highlighted that the configuration can affect the behavioral stability of the system and that it may not return to its original state post a sudden change in demand.

In summary, Minimally Intelligent Agent-Based manufacturing offers low overhead and extensibility, which are features that could fulfil the needs of Makerspaces. This was the rationale for considering them in the study. The studies reported demonstrated that the emergent behaviors of Minimally Intelligent Agent-Based manufacturing systems are not easy to predict and quantify *a priori*. This highlights the need for numerical studies that examine their utility in specific contexts and scenarios. The context and scenario of interest in this study are Makerspaces experiencing diverse non-repeating demand profiles.

Numerically modeling a Minimally Intelligent Agent-Based Makerspace

The numerical model was split into two elements. The first element represented the environments as a Minimally Intelligent manufacturing system. The second element formed the diverse non-

repeating demand profile. Each element is now described starting with the theory followed by the implementation.

A Minimally Intelligent Makerspace manufacturing system

The agent-based model featured two agent populations – Machine and Job – and a single Broker agent (Fig. 4). The Machine agents represented the AM machines and featured necessary information to represent the manufacturing capability. It was assumed that the machines could manufacture all of the jobs entering the system (i.e., Boolean checks would have been performed prior to the job entering the network, such as volume and material selection), any change-over time was constant, and the machines printed right first-time. No additional maintenance or checks were considered in the model. The Job agents represented the jobs that need to be manufactured and featured a job time that was defined randomly on creation. The Broker agent represented the network (i.e., cloud or local server) resources that would be required to maintain and facilitate communications and brokered connections and communication between the Machine and Job agent.² The Broker agent permitted direct and broadcast communication. Communication can be configured between Machine–Machine, Job–Job, Machine–Job, and Job–Machine.

No precedence exists between Machine and Job agents, making the system “queueless” with jobs representing a pool of work (Gopsill et al., 2022). The determination of which job will be manufactured by which machine is based on the minimal intelligences and communication strategy employed. The combination of machines, m , and Minimally Intelligent logics, n , affords considerable system configurability – $m \times n$. And it is the optimal configuration of the system for diverse demand that the study aimed to solve.

The model was implemented in Anylogic, which enabled a user to define the working pattern of the system (e.g., 9 am–5 pm) and the number of machines and logics, respectively. Five minimal intelligence logics were selected from existing job-scheduling research, and all could operate using only print time as the decision variable (Goudswaard et al., 2021). The logics were as follows:

- *First-Response First-Serve (FRFS)*: Selects the first job that replies to its request;
- *First-Come First-Serve (FCFS)*: Selects the job that was submitted earliest;
- *Longest Print Time (LPT)*: Selects the job with the longest print time;
- *Shortest Print Time (SPT)*: Selects the job with the shortest print time; and,
- *Random*: Randomly selects a job using a uniform distribution.

Figure 5a shows the main model view, which contains the parameters, agent populations, and metrics that were captured during the simulation.

Figure 5b shows the Job agent's logic flow. Job agents were created based on a pre-defined demand profile that was read in from a plain-text text file. On creation, Job agents enter the AVAILABLE state where they listen and respond to requests from Machine agents asking whether they are available. The agents also listen for messages from Machine agents to say whether they have selected the job, and if received, the Job agent moves to the SELECTED state. The Job agent will remain in this state until it

¹Please see Hui et al. (2021) for an example application exploiting cloud compute for smart scheduling.

²Please see Giunta et al. (2023) for more information on a real-world implementation.

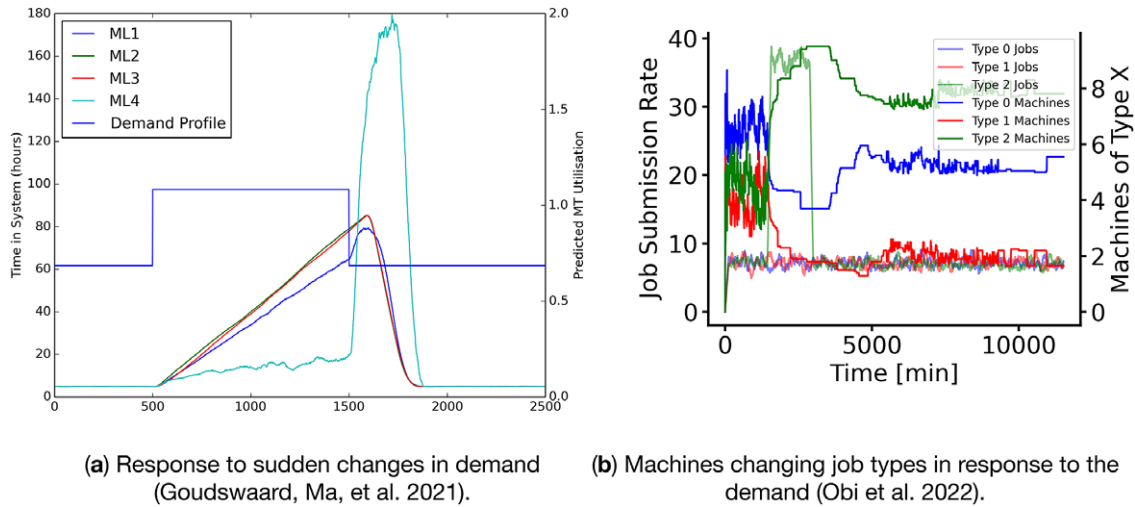


Figure 3. Studies into Minimally Intelligent Manufacturing Systems.

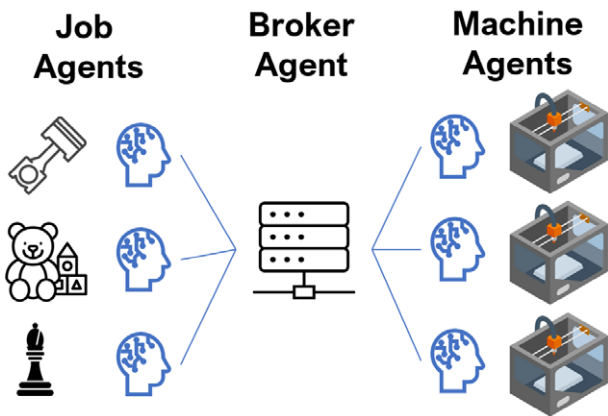


Figure 4. Agent-based model of a Minimally Intelligent Agent-Based manufacturing system.

receives a “complete” message from the Machine agent. On receiving the message, the Job agent moves to the COMPLETE state, which renders it inert for the rest of the simulation.

Figure 5c shows the Machine agent’s logic flow. The agent starts as AVAILABLE before proceeding to check whether the time is currently within a working day. If it is in the working day, the Machine agent broadcasts a message through the Broker agent to all Job agents asking if they are available. The Machine agent then waits for a pre-defined time for responses. The Machine agent then selects a Job agent from the response set based on its minimal intelligence and sends a “selected” message to the Job agent where it then waits to receive a confirmation. Confirmation is required for cases where another Machine agent may have selected the Job agent whilst the Machine agent had been deliberating. If the Job agent confirms the selection, the Machine agent moves to the manufacturing state and returns a “complete” message to the Job agent when finished. A typical communication pattern between Machines and Jobs is shown in Figure 6.

The model was validated in a previous study using a Living Lab empirical experiment that confirmed it approximated real-world operations (Giunta et al., 2023). With the model formed, section “Numerically modeling a Minimally Intelligent Agent-

Based Makerspace” continues to describe how a demand profile for diverse demand was created.

Modeling demand

As mentioned in the introduction and related work, the demand profile received by Makerspaces varied, was inconsistent, and rarely repeated themselves. To model this, the study exploited the unique characteristics of irrational numbers to seed the volume and inter-arrival time profiles of jobs entering the system.

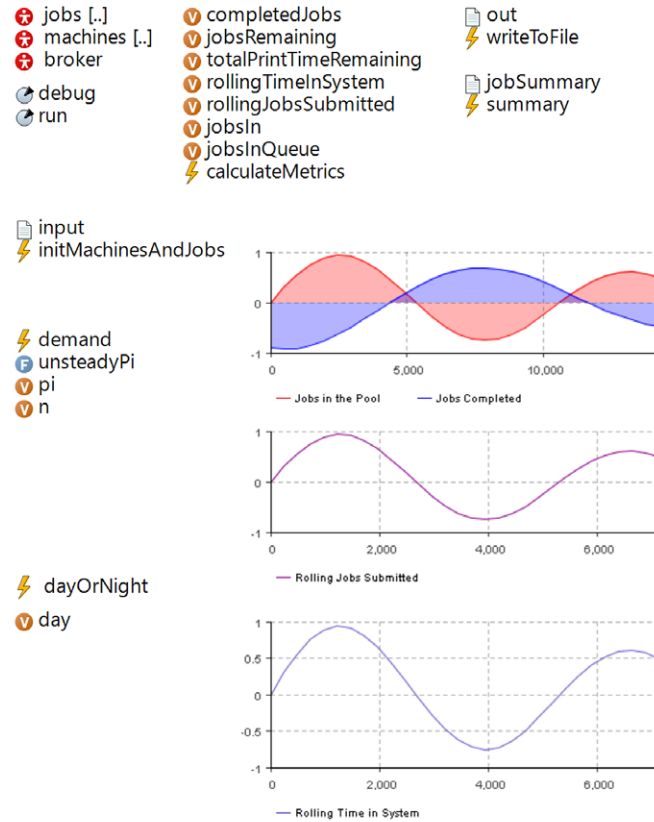
Irrational numbers provide an infinite non-repeating pattern of numerical values. Examples include π , ϵ , the golden ratio, and the square root of primes. Further, it is deterministic, which affords repeatability. Non-repeating patterns could also be achieved through pseudo-random values taken from a uniform distribution, but assurances must be made to use the same seed and implementation of the pseudo-random algorithm to achieve repeatability. Irrational numbers remove this dependency.

The method takes an irrational number and iterates through the sequence of values. The value is multiplied by a coefficient α_1 , resulting in a non-repeating inter-arrival time. A further irrational number (or another point in an irrational numbers’ sequence) is taken and iterated through to determine the volume of jobs. A coefficient α_2 is applied to scale the volume of jobs. Diversity in Job Time was modeled through a seeded probability distribution of manufacturing times from product history. The demand profile can then be tuned through α_1 and α_2 to provide the desired “loading” on a system.

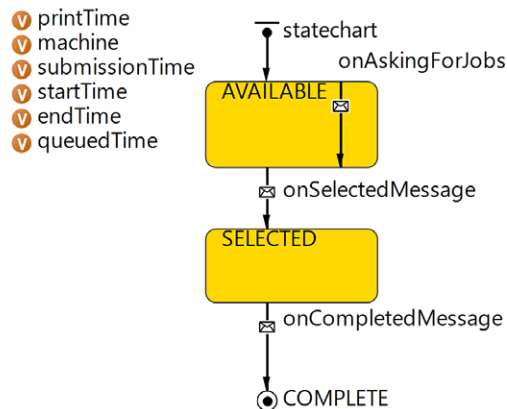
Figure 7 provides an example demand profile generated using π as the seed with index of 0 and 1,000 for the starting points for the inter-arrival time and volume; α_1 and α_2 were set to 1 and 20, respectively. The demand profile was generated using a Python script that outputted a pre-defined list of job submissions. This was then used in the agent-based simulation.

Examining Minimally Intelligent Makerspace manufacturing performance

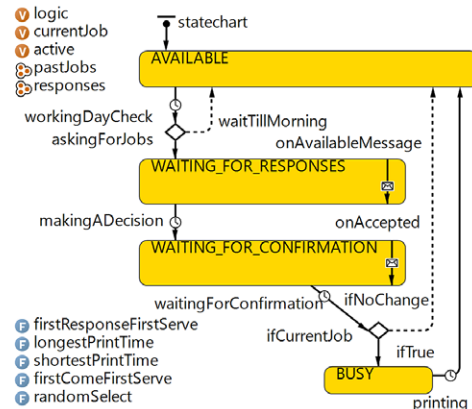
The study ranked the responsiveness of a Makerspace with 5, 10, 15, and 20 AM machines operating 9 am–5 pm. The machines were



(a) Main – Represents the global context in which the agents reside within and features the system performance variables and function for generating jobs through time.



(b) Job Agent – The variables and logic flow a job follows.



(c) Machine – The variables and logic flow a machine follows.

Figure 5. Images of the AnyLogic model used in the study.

configured with one of five logics – FRFS, FCFS, LPT, SPT, and Random selection – resulting in a full factorial study of 15,629 configurations. It was assumed all brokered jobs were printed successfully at their first attempt.

The demand profile was generated as a $f(\pi)$ for both the inter-arrival time and the volume of jobs, with α_1 and α_2 set to values detailed in Table 2 for each scale, respectively. Job manufacturing times were randomly selected from a triangular distribution whose lower, upper, and middle bounds were set to 48, 600, and

240 minutes, and represented typical print times for objects being submitted to a Makerspace and included the change-over/re-configuration time, which was considered constant for all jobs (Novak and Loy, 2020). The combined α_1 and α_2 values and job manufacturing time gave an approximate sum job print time in the pool of 94,000 min/machine across all scales.

The random selection from the triangular distribution used a fixed seed, enabling the profile to be re-created for future experiments. A copy of the demand profile can be found in

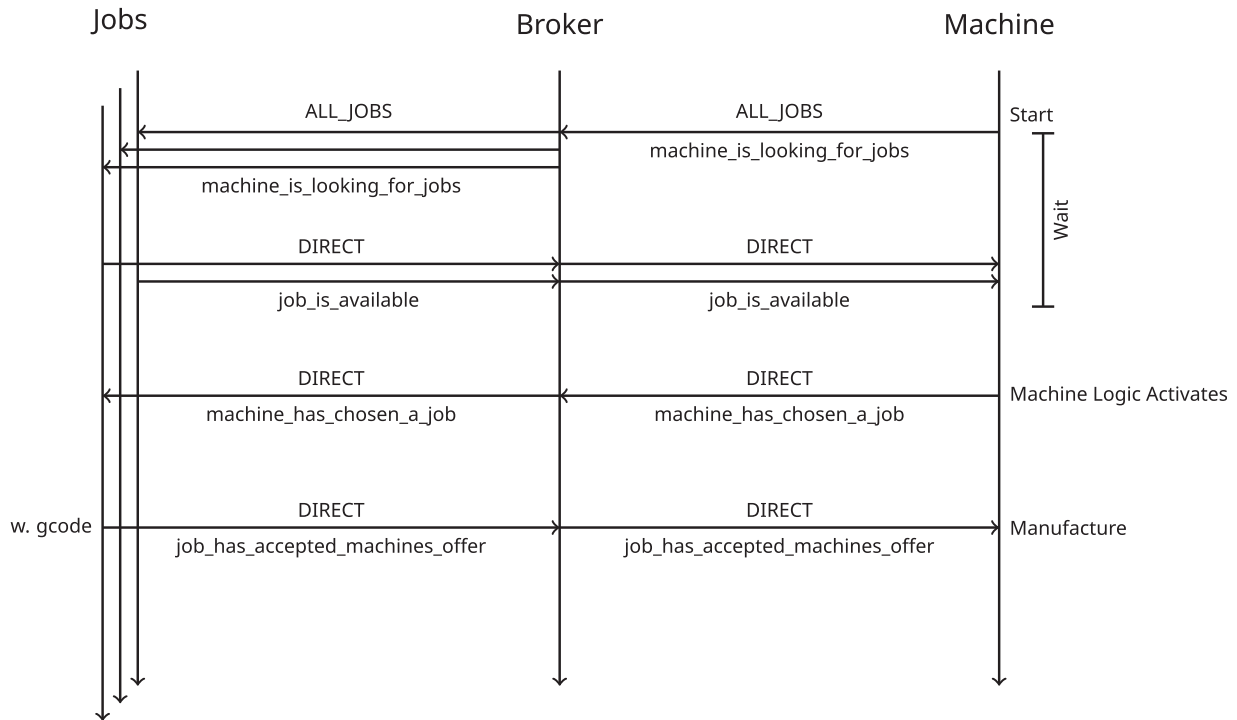


Figure 6. An example of the typical communication pattern between the Machine and Job agents.

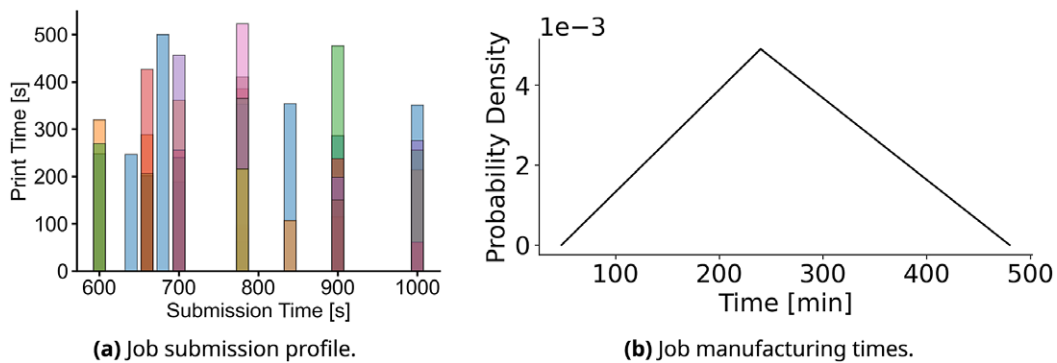


Figure 7. Modeling workshop demand.

Table 2. α settings to achieve the desired loading on the Makerspace

	Scale			
	5	10	15	20
α_1	20	20	20	20
α_2	0.4	0.7	1	1.3

https://github.com/jamesgopsill/mi_aiedam_model and is visualized in full in Figure 8.

The primary criterion used to assess responsiveness was Job Time-in-Pool (TiP). Time-in-Pool is the time a job spends in the pool “waiting” to be manufactured and is the difference between time in (t_i) and time out (t_o) minus the manufacturing time (t_m).

$$TiP = t_o - t_m - t_i. \tag{1}$$

Manufacturing time is subtracted as the system has no control over the feature. The minimum, median, mean, standard deviation, and maximum TiP values give an insight into how the system is responding to the demand. A high-performing system should minimize TiP such that jobs are shipping to the customers as quickly as possible.

To determine whether configurations consistently outperformed one another, a rank order according to ΣTiP of jobs that had been completed or were still in the pool was calculated after each simulated day. It can be considered equivalent to a football league table with the score being the ΣTiP . The edit distance (Deibel et al., 2005) – the number of moves and their distances up and down the ranking required to transform one ranking to another – between adjacent daily rankings was performed. This provided a

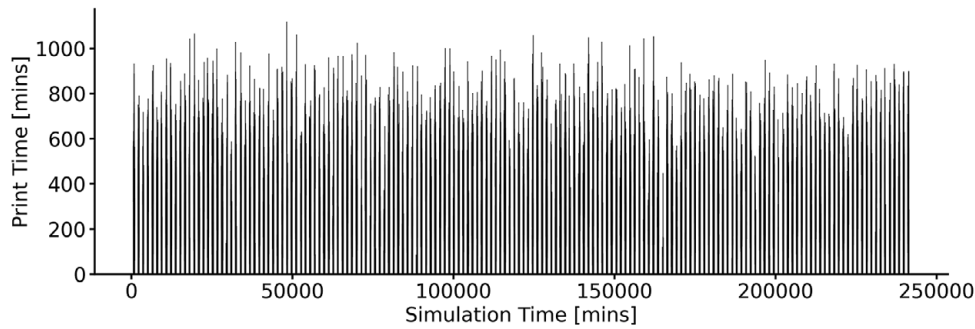


Figure 8. Diverse demand profile used in study.

global convergence criterion that was normalized against the largest change in the league table (i.e., the change between day 0 and day 1 where day 0 featured the configurations in a randomly generated order). Normalizing the edit distance produced a ratio that $\rightarrow 0$ when no change in the ranking occurs.

The global convergence was further supported by a local convergence criterion. The local convergence criterion evaluated the number of configurations entering and exiting the top and bottom 100 of the ranked list. The hypothesis was that performance across configurations would be normally distributed, and, as a result, the edit distance would never fully reach 0, with configurations regularly changing places with their neighbors who were close on performance. Thus, if this top 100 became consistent then one can confidently claim that the set of top-performing configurations had been determined.

Having checked for convergence, an analysis of the system, machine, and job behavior was performed. The system behavior was examined via the distribution of logics across 5% ranked percentiles with the hypothesis that some logics may appear more prominently in the most and the least responsive systems. The distribution of Σ TiP, messages sent, number of rejections, and time spent printing across all the system configurations were also analyzed.

Machine agent behavior was analyzed by taking the most and least responsive systems and evaluating individual machine utilization. The hypothesis was that the distribution of work across the machines would be different for most and least responsive system configurations.

Job agent behavior was analyzed through histograms of TiP for the most and least responsive systems. Job print times and

submission times were then correlated with the TiP as it was hypothesized that some system configurations would favor particular jobs over others.

Results

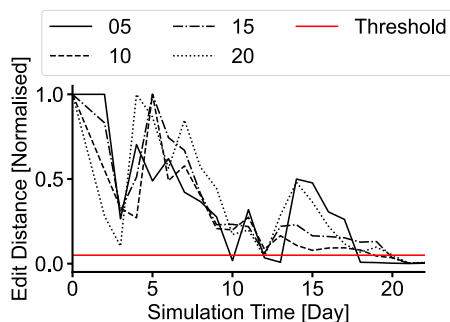
The study was run on a dual 12-core Intel Xeon 256GB RAM workstation, took 3 hrs to complete, and resulted in a 4GB data log. Section “Convergence” presents the convergence results across the systems and sections “System behavior,” “Machine behavior,” and “Job behavior” present the system behavior from the perspective of a system, jobs, and machines, respectively.

Convergence

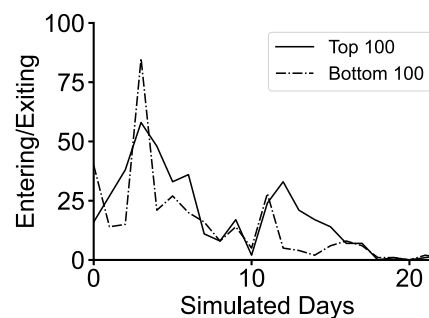
Figure 9a shows the convergence to a ranked list of responsive configurations for all scales. The edit distance scores fluctuate for the first few simulated days before dropping suddenly at day 10. This behavior was consistent across all system sizes.

Beyond day 10, the edit distance remains low, confirming the ranking has moved to a steady-state condition with configurations trading places with nearby ranks rather than distant ranks. There was a rise in the 5-machine league table at day 14 and 18 marks, suggesting the behavior of the system is more chaotic and easily perturbed favoring different configurations from day to day. Larger systems offer more consistency in their behavior from day to day.

Figure 9b shows the local convergence of the top and bottom 100 for the 15 machine set of configurations. The results corroborate the global convergence of the ranked list and shows by day 20 that little to no changes to the top and bottom 100 are made.



(a) Global Convergence



(b) Local Convergence (15 machine only).
N.b., all scales exhibited similar behaviour.

Figure 9. Convergence to a ranked list of performant configurations.

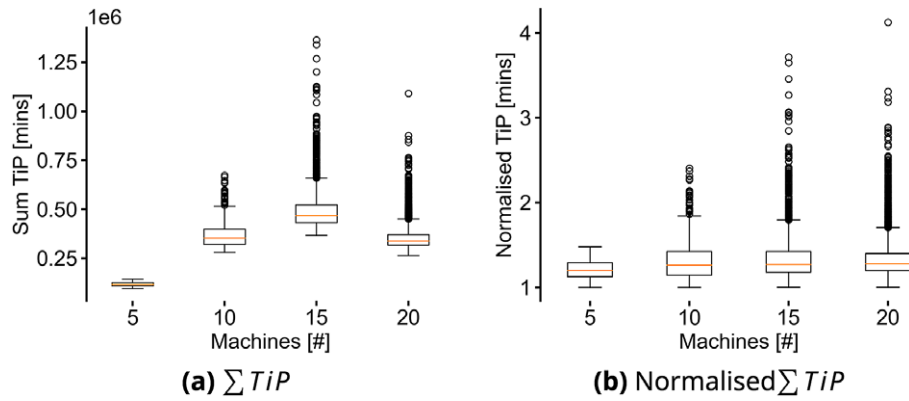


Figure 10. Responsiveness at scale.

Therefore we can be confident that we ascertained the most and least responsive configurations for a set of Minimally Intelligent machines experiencing diverse demand.

System behavior

Figure 10 shows the responsiveness of the system with respect to scale based on ΣTiP . Figure 10a shows there is a steady trend of increasing TiP as the size of the system increases. Figure 10b shows TiP normalized to the most responsive configuration for each scale. The normalization reveals that the systems exhibit the same underlying behavior and distribution of responsiveness, albeit with an increasing least-performing configuration tail as size increases. This suggests a relatively scale-free behavior for Minimally Intelligent Manufacturing Systems. The range of responsiveness increases beyond two- and three-fold for the large systems and demonstrates that greater gains/losses in responsiveness can be achieved as scale increases. The median is within 25% of the most responsive configuration and suggests that Makerspaces randomly selecting a configuration are likely to perform reasonably well.

Figure 11 provides matrix plots of the converged league tables for each system size. The configurations were grouped by

5 percentile and the ratio of logics presented across the configurations plotted. The larger systems (15 and 20) are more consistent in terms of system configuration trends, with the most responsive configurations consisting of mostly Random logic and a selection of other logics. The least-performing configurations consisted of no Random logics and a high number of FCFS logics.

The smaller system sizes (5 and 10) are more chaotic in their configuration of logics, making it harder to form any general heuristics. Nonetheless, the most responsive configurations tended to consist of a breadth of logics, while the worst performing configurations featured few to no Random selection logics.

Table 3 details the five most and least responsive configurations for each of the system scales. There is a consistency across the scales for most responsive configurations to feature a variety of Minimally Intelligent logics. There is also a trend to increase the number of Random logics as system scale increases. The least responsive system commonly feature a single logic which is consistently LPT or FCFS.

Figure 12 plots system behavior in terms of the messages sent, rejections, and time spent printing. Figure 12a shows that there is a steady increase in the message count as the configuration moves down the ranking followed by a sharp increase at the least responsive configurations. This is the same for all cases but less apparent

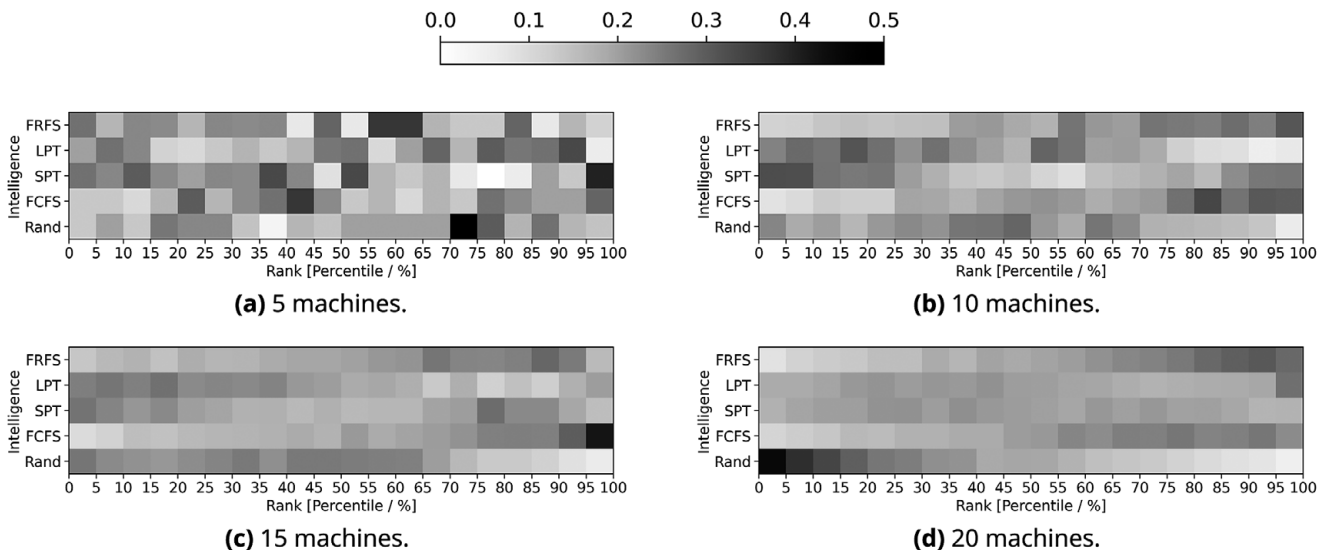


Figure 11. League tables represented as a matrix and aggregated in 5% percentiles. The colorbar summarizes the number of machines using the logics.

Table 3. The most and least responsive configurations across the system scales

#	Scale (# Machines)																			
	5					10					15					20				
	FRFS	LPT	SPT	FCFS	RAN	FRFS	LPT	SPT	FCFS	RAN	FRFS	LPT	SPT	FCFS	RAN	FRFS	LPT	SPT	FCFS	RAN
1	2	1	1	1	0	1	3	1	0	5	2	3	6	0	4	0	4	4	0	12
2	0	0	1	1	3	0	3	3	0	4	0	2	3	6	4	0	5	4	1	10
3	0	3	2	0	0	3	5	2	0	0	5	3	5	0	2	0	5	1	2	12
4	2	1	2	0	0	0	2	3	1	4	2	2	4	0	7	1	5	3	0	11
5	3	1	1	0	0	0	3	5	1	1	0	3	5	0	7	1	5	4	2	8
...																				
-5	1	0	4	0	0	0	0	9	1	0	1	14	0	0	0	2	18	0	0	1
-4	3	0	0	2	0	0	1	8	0	1	0	15	0	0	0	0	19	0	1	0
-3	0	0	4	1	0	0	10	0	0	0	0	0	0	15	0	0	19	0	0	1
-2	0	0	0	5	0	0	0	0	10	0	0	0	0	14	0	1	19	0	0	0
-1	0	0	5	0	0	0	1	9	0	0	0	14	0	1	0	0	20	0	0	0

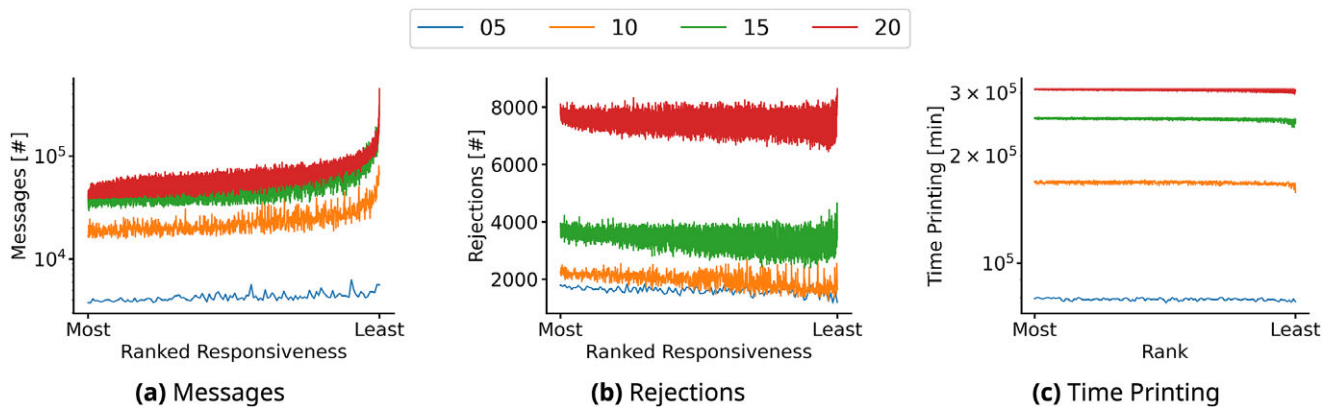


Figure 12. System behavior through the lens of messages sent, job-machine rejections, and time spent printing.

for 5 machines. It was noted that message count power-law scales with system size (N.b., use of log-scale on the y-axis).

Figure 12b shows a relatively even number of rejections across the configurations for all system sizes with a slight decline when reaching the least responsive configurations and then an abrupt increase for the very least. The variance in rejections also increases when moving from the most to least responsive configurations.

Figure 12c plots the sum time spent printing across the machines with the most responsive configurations spending the most time printing, with a slow decline in time printing before a sudden drop for the least responsive configurations, although the log-scale does minimize the impact of the drop. The steady nature of the time printing shows that all the systems are busy processing jobs. Therefore, it is the order in which they are processing the jobs that the configurations have control over and enables the responsiveness of the system to be manipulated.

Machine behavior

Figure 13 compares the machine utilization for the most and least responsive configurations in responding to the diverse demand profile. The utilization has been normalized against the sum of

the simulated working day hours. Both the most and the least responsive configurations feature machines that operate beyond the working day (i.e., machines have managed to select a job to continue printing into the night). In the least responsive configurations, all machines work beyond the working day, while the most responsive configurations see the SPT machines working 9 am–5 pm. This is a logical output as small job times are unlikely to take the machine late into the evening.

The least responsive machines feature a relatively even loading across their machines. In comparison, the most responsive configurations feature machines that experience much greater utilization than others. This is correlated with the logic placed on the machine, with LPT machines working much longer than their SPT and RAN partners. This is, again, logical as they are likely to select jobs that can take them long into the night.

Job behavior

Figure 14 shows the job TiP distributions for the most and least responsive configurations. The least responsive configurations are penalized heavily by having jobs that stay in the pool for multiple days. In all cases, there are jobs that are waiting in the system for

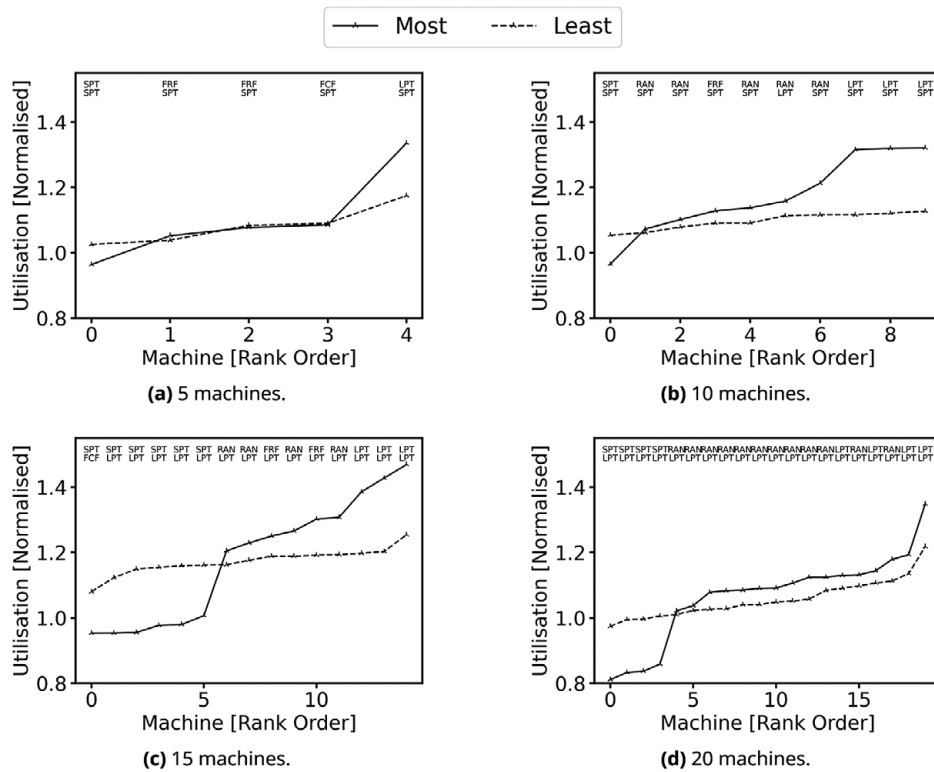


Figure 13. The individual machine utilization for the most and least responsive system at the different scales. The machines have been placed in rank utilization order and the logic being used by the machine detailed at the top each figure. The top set of logics in each figure refer to the most responsive configuration and the bottom set to the least responsive configuration.

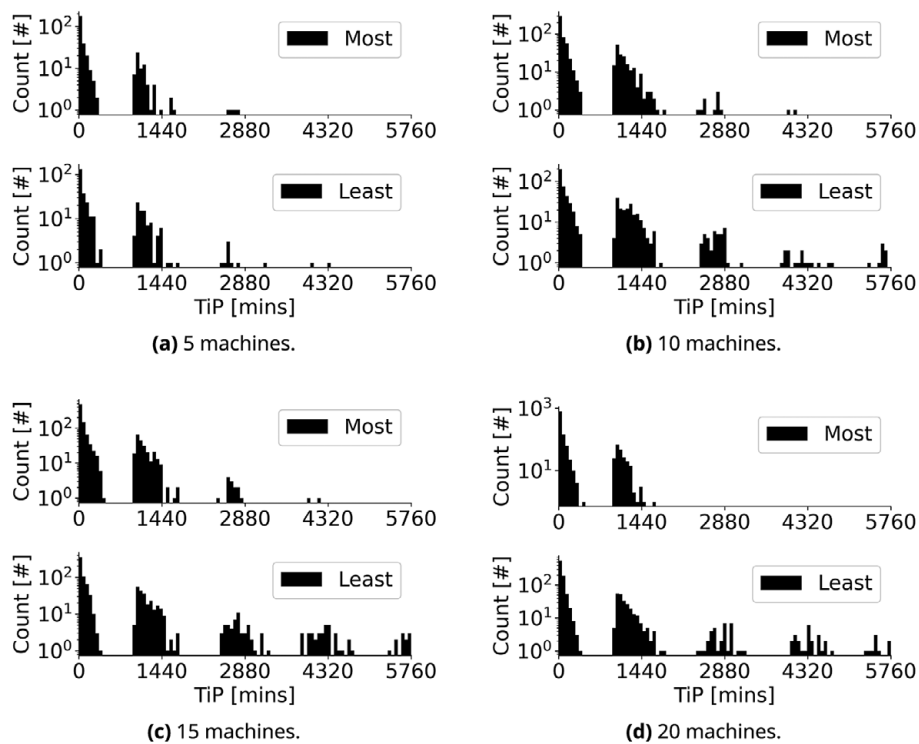


Figure 14. Job TiP Distribution across the system scales. Notice the steps in the distribution indicating that a day has passed where the job has been waiting and that the tail extends across multiple days for the least responsive configurations.

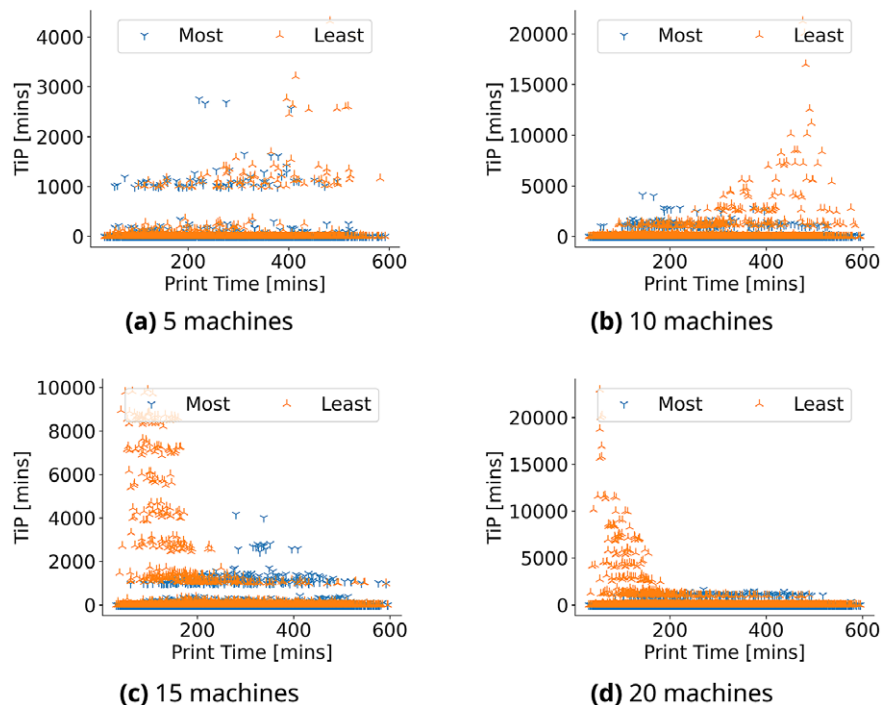


Figure 15. Print time correlation.

more than 5 days. The most responsive configurations are able to complete the majority of jobs within a day of being submitted.

Figure 15 shows the correlation between job print time and TiP for the most and least responsive configurations. The most responsive configurations show little to no bias towards jobs of particular durations and are consistent across the scales studied.

In contrast, the least responsive configurations show bias, and the bias is different across the system scales. Five and 10 machines scales favor small print time jobs, while 15 and 20 machine jobs favor longer print time jobs. This is in agreement with the configuration of logics as the worst performing configurations for 15 and 20 machines, which feature many LPT logics. The LPT logics focus on the long print time jobs at the detriment of the shortest print jobs.

Figure 16 shows the correlation between submission time and TiP. No correlation was observed across the system scales, suggesting there is no preferential treatment for jobs being submitted at particular times of the day.

Discussion

The results show that Minimally Intelligent Agent-Based manufacturing system configurations converge to a steady-state ranking and the convergence occurs relatively quickly (approx. 10–20 simulated days). This is interesting as it shows that a Makerspace experiencing a diverse demand can select a single optimal configuration and the system will continue to operate in an optimal operating window thereafter.

The results have also shown that configuration performance can vary by up to 200%. If an environment were to randomly select a configuration then it would likely perform 25–50% from the optimal configuration. The result highlights the benefit of numerical studies in providing an appropriate list of configurations for

Makerspace operators wishing to take a Minimally Intelligent Agent-Based manufacturing systems approach.

The results also showed a tendency for TiP to increase with system size (scale). This is likely due to the increased chance for machines to bid for the same work, thus resulting in a greater number of rejections, which adds delay to the system. Normalizing the result showed that the behavior is scale invariant across the configurations and suggests that results from small-scale system studies could inform and support the operation of larger-scale systems.

The league tables and percentile matrices (Table 3 and Fig. 11) also confirmed this behavior with the most responsive configurations featuring a variety of logics, with the proportion of Random logic machines increasing inline with system scale. The reasoning is that featuring a variety of logics ensures machines are targeting different areas of the demand profile and, thus, are less likely to compete with one another. Random logic further reduces the likelihood of machines targeting the same job during the bidding process. In comparison, the least responsive configurations feature predominately single logics, LPT and FCFS in particular, that would result in machines competing with one another and, thus, slowing down the response of the system. Therefore, for operators of new Makerspaces, having machines with a range of logics is a good starting position and machines that are added over time should automatically start with Random selection logic.

The observed range in performance (Fig. 10) can be attributed to the significant penalty placed on a job if it goes beyond multiple days of not being manufactured. This is a result of the 9 am–5 pm working pattern and confirmed by the job distributions with the least responsive configurations having a long-tail of jobs with a 3–5 day wait.

Further, the FCFS approach commonly employed by Makerspaces is analogous to an all FCFS configuration. This configuration consistently performed poorly and featured at the bottom of the

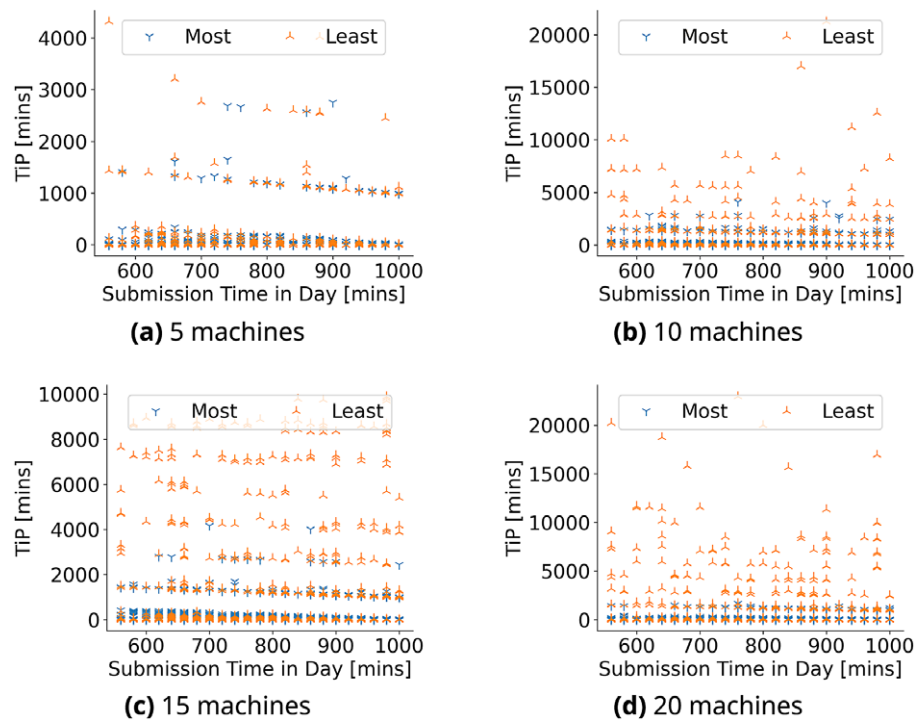


Figure 16. Submission time correlation.

league tables (Table 3 and Fig. 11). Therefore, many Makerspaces are currently operating far from their full capacity or maximal responsiveness. It also highlights that almost any configuration of Minimally Intelligent agents would offer an improvement over existing practice.

An interesting feature of the system dynamics is that both the most and the least responsive configurations feature a high number of job rejections. This highlights that machines are bidding for the same work with one machine winning and the others losing out. In the case of the most responsive configurations, it is likely that the job pool was near-empty and the machines were scrabbling over the few remaining jobs. In the case of the least responsive configurations, the machines were likely identifying and bidding for the same job at the same time, even though there are plenty of other jobs in the pool.

The machine utilization results (Fig. 13) highlighted a disparity in utilization across the machines in the most responsive configurations. This would be important to monitor as the machines will likely degrade differently over time so the operators may likely need to swap machine logics to balance the loading to extend the operating life of the system without incurring maintenance. This would be an interesting extension to this work and might further explore whether agents could communicate and make decisions by swapping their selection logics to extend system operating life.

The job behavior analysis (section “Job behavior”) revealed that the most responsive configurations featured no correlation between job parameters, print time and submission time, and likely time in the pool. This is beneficial to individuals submitting to the Makerspaces as their job will not be unfairly disadvantaged by any of these characteristics. In contrast, the least responsive configurations introduced bias, with short print time jobs being disadvantaged as the machines were focused on delivering the longer print time jobs.

Limitations of the work are bounded by the scale of system studied, the data used to evaluate decisions, and the logics considered. The trend of featuring more Random logics needs to be verified through simulations of systems containing more machines. The data used was purely print time, and constraints, such as the availability of filament, filament types, multi-material printing, and print volumes, would need to be taken into account in more complex Makerspace environments. The model also assumed perfect printing the first time around, which may not always be the case. It would be interesting to add and vary print success rates to understand the trade-offs between improving right first-time printing and the optimal configuration of logics of a particular demand profile.

Five logics were evaluated in this study, and there could be many others that could feature on these machines that could further improve the performance of the system; for example, where machines are aware of the time of day and, as a result, bid for different types of job, and the ability to batch select and queue up multiple jobs for the machine the AI agent is representing. Further trade-offs between optimally operating an existing set of machines and/or simply purchasing more machines could also be considered as well as how different job print distributions could affect which configurations are optimal.

Future work could investigate how a dynamically changing set of machine logics could provide further improve Σ TiP. Future work could also build on the results, including expanding the analysis to consider negotiation protocols beyond coordinated manufacturing where the job has a larger role in the decision-making process. This may be required in scenarios where users wish to submit to one of many workshops that are managed by different firms and potentially in competition with one another. Also, while the model was validated through a Living Lab experiment, more work could be done to capture and monitor workshop practice to provide empirical datasets to test and validate models against.

Conclusion

Makerspaces, Hackspaces, FabLabs, and Work/job-shops are essential services supporting many aspects of Design & Manufacture (D&M) from education and consumer items through to prototyping the next innovation and producer of critical components in hard-to-reach locations and developing countries. In each case, there is a need to minimize the operational costs and maximize the capability of the suite of AM machines being deployed to improve responsiveness and throughput of revenue-generating work.

This article has reported a study that has defined a Minimally Intelligent Agent-Based model for Makerspace environments and ran a full factorial analysis of 5, 10, 15, and 20 Minimally Intelligent Agent AM machines. Optimally configured set of machines can achieve a 200% improvement in responsiveness. Also, all FCFS configurations – a proxy to current practice – consistently feature in the least responsive configurations. The optimal configurations for 5, 10, 15, and 20 were reported and can be used by Makerspaces to significantly improve their performance. Optimal configuration machine utilization is not evenly distributed across the machines, resulting in machines possibly degrading at different rates.

Competing interest. The authors declare none.

Funding statement. The work has been undertaken as part of the Engineering and Physical Sciences Research Council (EPSRC) grants – EP/R032696/1, EP/V05113X/1, and EP/W024152/1.

Data availability statement. The data that support the findings of this study are openly available at https://github.com/jamesgopsill/mi_aiedam_model.

References

- Aidala S, Eichenberger Z, Chan N, Wilkinson K and Okwudire C (2022) MTouch: An automatic fault detection system for desktop FFF 3D printers using a contact sensor. *The International Journal of Advanced Manufacturing Technology* 120(11–12), 8211–8224.
- Annear LM, Akhavan-Tabatabaei R and Schmid V (2023) Dynamic assignment of a multi-skilled workforce in job shops: An approximate dynamic programming approach. *European Journal of Operational Research* 306(3), 1109–1125. <https://doi.org/10.1016/j.ejor.2022.08.049>.
- Anon (2023a) *Home page*. Available at <https://fabfoundation.org> (accessed 6 August 2023).
- Anon (2023b) *Home page*. Available at <https://www.repaircafe.org/en/> (accessed 6 August 2023).
- Basuki B, Yoto AS and Tjiptady BC (2020) Management model of manufacturing work shop/Laboratory of vocational education in the industry 4.0. In *2020 4th International Conference on Vocational Education and Training (ICOVET)*. Malang, Indonesia: IEEE, pp. 127–130. <https://doi.org/10.1109/ICOVET50258.2020.9230188>.
- Baumann F and Roller D (2016) Vision based error detection for 3D printing processes. *MATEC Web of Conferences* 59, 06003.
- Chen T-CT (2019) Fuzzy approach for production planning by using a three-dimensional printing-based ubiquitous manufacturing system. *AI EDAM* 33(4), 1–11. <http://doi.org/10.1017/S0890060419000222>.
- Chung I-Y and Cheol-HeeYoo S-JO (2013) Distributed intelligent microgrid control using multi-agent systems. *Engineering* 5(1), 1–6. <http://doi.org/10.4236/eng.2013.51b001>.
- Cliff D (1997) *Minimal-Intelligence Agents for Bargaining Behaviors in Market-Based Environments*. Hewlett-Packard Labs Technical Reports.
- Daoulas T, Bizaoui V, Dubrana F and Francia RD (2021) The role of three-dimensional printing in coronavirus disease-19 medical management: A French nationwide survey. *Annals of 3D Printed Medicine* 1, 100001. <https://doi.org/10.1016/j.stlm.2020.100001>.
- Deibel K, Anderson R and Anderson R (2005) Using edit distance to analyze card sorts. *Expert Systems* 22(3), 129–138. <http://doi.org/10.1111/j.1468-0394.2005.00304.x>.
- Eyers D (2018) Control architectures for industrial additive manufacturing systems. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 232(10), 1767–1777.
- Ford S and Despeisse M (2016) Additive manufacturing and sustainability: An exploratory study of the advantages and challenges. *Journal of Cleaner Production* 137, 1573–1587. <https://doi.org/10.1016/j.jclepro.2016.04.150>.
- Giunta L, Hicks B, Snider C and Gopsill J (2023) A living lab platform for testing additive manufacturing agent-based manufacturing strategies. *Procedia CIRP* 119, 656–661. <https://doi.org/10.1016/j.procir.2023.03.118>.
- Giunta L, Obi M, Goudswaard M, Hicks B and Gopsill J (2022) Comparison of three agent-based architectures for distributed additive manufacturing. *Procedia CIRP* 107, 1150–1155. <https://doi.org/10.1016/j.procir.2022.05.123>.
- Gopsill J and Hicks B (2016) The issues & challenges of 3D managed print services: Towards a support tool for 3D managed print services. *InImpact: The Journal of Innovation Impact* 8(2), 478.
- Gopsill JA and Hicks BJ (2018) Investigating the effect of scale and scheduling strategies on the productivity of 3D managed print services. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 232, 1753–1766. <http://doi.org/10.1177/0954405417708217>.
- Gopsill J, Obi M, Giunta L and Goudswaard M (2022) Queueless: Agent-based manufacturing for workshop production. In Jezic G, Chen-Burger Y-HJ, Kusek M, Šperka R, Howlett RJ and Jain LC (eds.), *Agents and Multi-Agent Systems: Technologies and Applications 2022*. Singapore: Springer Nature Singapore, pp. 27–37. https://doi.org/10.1007/978-981-19-3359-2_3.
- Goudswaard M, Kent L, Giunta L, Gopsill J, Snider C, Valjak F, Christensen KA, Felton H, Ege DN, Real RM, Cox C, Horvat N, Kohtala S, Eikevåg SW, Martinec T, Perišić MM, Steinert M and Hicks B (2022) Virtually hosted hackathons for design research: Lessons learned from the international design engineering annual (IDEA) challenge 2021. *Proceedings of the Design Society* 2, 21–30. <https://doi.org/10.1017/pds.2022.3>.
- Goudswaard M, Ma A, Nassehi A, Hicks B and Gopsill J (2021) Responding to rapidly changing product demand through a coordinated additive manufacturing production system: A COVID-19 case study. *IOP Conference Series: Materials Science and Engineering* 1193, 012119. <https://doi.org/10.1088/1757-899x/1193/1/012119>.
- Heragu S, Graves R, Kim B-I and St Onge A (2002) Intelligent agent based framework for manufacturing systems control. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 32, 560–573. <https://doi.org/10.1109/TSMCA.2002.804788>.
- Hui J, Lei J, Ding K, Zhang F and Lv J (2021) Autonomous resource allocation of smart workshop for cloud machining orders. *AI EDAM* 35(2), 226–239. <http://doi.org/10.1017/S089006042000044X>.
- Lomas K and Cliff D (2020) Exploring narrative economics: An agent-based-modeling platform that integrates automated traders with opinion dynamics. *arXiv preprint arXiv:2012.08840*.
- Longhitano GA, Nunes GB, Candido G and da Silva JVL (2021) The role of 3D printing during COVID-19 pandemic: A review. *Progress in Additive Manufacturing* 6(1), 23639520. <http://doi.org/10.1007/s40964-020-00159-x>.
- Lu Y, Xu X and Xu J (2014) Development of a hybrid manufacturing cloud. *Journal of Manufacturing Systems* 33(4), 551–566. <https://doi.org/10.1016/j.jmsy.2014.05.003>.
- Ma A, Nassehi A and Snider C (2021) Anarchic manufacturing: Implementing fully distributed control and planning in assembly. *Production & Manufacturing Research* 9(1), 56–80. <https://doi.org/10.1080/21693277.2021.1963346>.
- Mersand S (2021) The state of makerspace research: A review of the literature. *TechTrends* 65(2), 174–186.
- Nagle SB (2021) Maker services in academic libraries: A review of case studies. *New Review of Academic Librarianship* 27(2), 184–200. <http://doi.org/10.1080/13614533.2020.1749093>.
- NESTA (2015) *Open dataset of UK makerspaces*.
- Novak JI and Loy J (2020) A quantitative analysis of 3D printed face shields and masks during COVID-19. *Emerald Open Research* 2, 42. <http://doi.org/10.35241/emeraldopenres.13815.1>.
- Obi M, Snider C, Giunta L, Goudswaard M and Gopsill J (2022) Coping with diverse product demand through agent-led type transitions. In Jezic G, Chen-Burger Y-HJ, Kusek M, Šperka R, Howlett RJ, and Jain LC (eds.), *Agents and Multi-Agent Systems: Technologies and Applications 2022*. Singapore: Springer Nature Singapore, pp. 277–286.

- Oh Y, Witherell P, Lu Y and Sprock T** (2020) Nesting and scheduling problems for additive manufacturing: A taxonomy and review. *Additive Manufacturing* **36**, 101492. <https://doi.org/10.1016/j.addma.2020.101492>.
- Pantoja CE, Soares HD, Viterbo J and Seghrouchni AEF** (2018) An architecture for the development of ambient intelligence systems managed by embedded agents. In *SEKE*. KSI Research Inc, p. 215. <https://ksiresearch.org/>.
- Paraskevoudis K, Karayannis P and Koumoulos EP** (2020) Real-time 3D printing remote defect detection (stringing) with computer vision and artificial intelligence. *PRO* **8**(11), 1464.
- Petsiuk AL and Pearce JM** (2020) Open source computer vision-based layer-wise 3D printing analysis. *Additive Manufacturing* **36**, 101473.
- Priore P, De La Fuente D, Gomez A and Puente J** (2001) A review of machine learning in dynamic scheduling of flexible manufacturing systems. *AI EDAM* **15**(3), 251–263. <http://doi.org/10.1017/S0890060401153059>.
- Priore P, Gómez A, Pino R and Rosillo R** (2014) Dynamic scheduling of manufacturing systems using machine learning: An updated review. *AI EDAM* **28**(1), 83–97. <http://doi.org/10.1017/S0890060413000516>.
- Pryor S** (2014) Implementing a 3D printing Service in an Academic Library. *Journal of Library Administration* **54**(1), 1–10. <http://doi.org/10.1080/01930826.2014.893110>.
- Purusothaman SRRD, Rajesh R, Bajaj KK and Vijayaraghavan V** (2013) Implementation of arduino-based multi-agent system for rural Indian micro-grids. In *2013 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia)*. Bangalore, India: IEEE, pp. 1–5. <https://doi.org/10.1109/ISGT-Asia.2013.6698751>.
- Rautray P and Eisenbart B** (2021) Additive manufacturing – Enabling digital artisans. *Proceedings of the Design Society* **1**, 323–332. <https://doi.org/10.1017/pds.2021.33>.
- Scalfani VF and Sahib J** (2013) A model for managing 3D printing services in academic libraries. *Issues in Science and Technology Librarianship* **72**, 1–13.
- Schonwetter T and Van Wiele B** (2018) 3D printing: Enabler of social entrepreneurship in Africa? The roles of FabLabs and low-cost 3D printers. In *Cape Town, South Africa: Open African Innovation Research (Open AIR)*. [Google Scholar]. OpenAIR, <https://openair.africa/3d-printing-enabler-of-social-entrepreneurship-in-africa-the-roles-of-fablabs-and-low-cost-3d-printers/>
- Smith P and Mortati M** (2017) Commons people: Additive manufacturing enabled collaborative commons production (by design). *International Journal of Rapid Manufacturing* **6.2–6.3**, 197–213. <https://doi.org/10.1504/IJRAPIDM.2017.082159>.
- Ultimaker** (2023) *Introducing Ultimaker Digital Factory*. Available at <https://ultimaker.com/learn/introducing-ultimaker-digital-factory/> (accessed 6 August 2023).
- Wilczynski V** (2015) Academic maker spaces and engineering design. In *2015 ASEE Annual Conference & Exposition*. American Society for Engineering Education, pp. 26–138. <https://doi.org/10.18260/p.23477>.