**Proceedings of the ASME 2018 International Design Engineering Technical Conferences
Computers and Information in Engineering Conference (IDETC/CIE 2018)
August 26-29, 2018, Quebec City, Canada, USA**

DETC2018-85801

DESIGNING MARKET THICKNESS AND OPTIMAL FREQUENCY OF MULTI-PERIOD STABLE MATCHING IN CLOUD-BASED DESIGN AND MANUFACTURING

J D Thekinen, Yupeng Han, Jitesh H. Panchal
Design Engineering Lab at Purdue (DELP)
School of Mechanical Engineering
Purdue University
West Lafayette, Indiana 47907

ABSTRACT

A central issue in two-sided matching markets such as Cloud-Based Design and Manufacturing (CBDM) where agents interact over a long period of time is the design of optimal matching period during recursive implementation. Existing literature provides mechanisms that satisfy useful properties such as stability in a single matching cycle, but they lack studies on the effect of the period of matching cycle on the optimality. To address this gap, we perform simulation studies on a synthetic CBDM scenario where service seekers arrive as a Poisson process with a fixed number of service providers offering resources. We identify the optimal matching period and assess its robustness using sensitivity studies. Optimality is measured in terms of utility obtained by the agents, the number of matches and fairness of the utility distribution. We show that a matching period equal to the ratio of the number of service providers to the arrival rate of service seekers is optimal.

NOMENCLATURE

S Set of service seekers.
P Set of service providers.
$|S|$ Number of service seekers.
$|P|$ Number of service providers.
s_{-i} Set of all service seekers excluding service seeker s_i.
p_{-j} Set of all service providers excluding service provider p_j.
R_{s_i} Preference ordering of service seeker s_i over its alternatives.
R_{p_j} Preference ordering of service provider p_j over its alternatives.
$j \succeq_i k$ i prefers j to k.
$j \succ_i k$ i strictly prefers j to k.
$M(s_i)$ Service provider to which service seeker s_i is assigned through matching mechanism M.
X Set of attributes.
$f(X)$ Single attribute utility towards attribute X.
u_{ij} utility attained by agent i being matched to alternative j.
w_{ki} Weight of attributes X_k for agent i.
t Time period of each matching cycle (in days).
t_k time of implementation of k^{th} matching cycle.
$M_{s_i}(t_k)$ Service provider to which service seeker s_i was matched in $k^t h$ matching cycle.
$M_{p_j}^{-1}(t_k)$ Service seekers being served by service provider p_j in $k^t h$ matching cycle.
T Total time duration during which matching mechanisms will be implemented (in days).
τ_{ij} Time to process service request of service seeker s_i by service provider p_j
$\overline{\tau}$ Average service processing time
λ Mean arrival rate of service seekers.
μ_j Mean service processing time of service provider p_j.
h_j Working hours of service seeker p_j per day.
$|M|$ Total number of stable matches possible under mechanism M.

1 Introduction

Cloud-Based Design and Manufacturing (CBDM) refers to a product realization model that enables collective open innovation and rapid product development with minimum costs through a social networking and negotiation platform between service providers and consumers [1]. One of the key characteristics of CBDM is on-demand self-service where users such as designers interact with other users such as manufacturers on a self-service basis. Thus CBDM involves interaction between two groups of participants: service seekers (or designers) and service providers (or manufacturers). *Service seekers* need to manufacture or use computational resources, but do not possess the capabilities to do so. *Service providers* own and operate equipment or other resources and are ready to offer users instantaneous access to these capabilities. CBDM needs to provide a platform to facilitate such interactions.

Conventional resource allocation methods are inappropriate for matching resources to service seekers in decentralized scenarios because the designer of the algorithm makes implicit assumptions that the participating agents will act as instructed. With the emergence of distributed and cloud-based manufacturing with independent resource providers, this assumption is no longer valid. It is more reasonable to assume that each participating agent will manipulate preferences for selfish gains at the cost of efficiency of the mechanism. The field of 'mechanism design'studies on designing algorithms where agents act rationally. The misrepresentation of information by individuals (or group of) is called 'strategic behavior'and the mechanisms that penalize such behavior are said to be 'strategic-proof' [2]. Efficiency loss because of strategic behavior need not be small and the cost of not having a strategy-proof mechanism is much harder to measure.

In addition to strategy-proofness, other useful properties of mechanisms include stability [3], individual rationality [4], consistency [5], monotonicity with respect to demand [6], and supply [7]. There is no mechanism that satisfies all these properties [4]. Therefore, mechanism must be specifically designed for each application. In an earlier work, the authors, modeled the problem of resource allocation in CBDM as a bipartite matching problem [8] and proposed best matching mechanisms [9] among existing mechanisms in different CBDM scenarios by analyzing their properties in the context of requirements in different CBDM scenarios. However, in a stochastic environment like CBDM where the arrival of service seekers and availability of service providers is a continuous process there are no studies on the optimal frequency of implementation of such mechanisms. Therefore, additional advances need to be made on existing mechanism design literature to develop optimal matching mechanism suited for the requirements of CBDM.

The job scheduling literature focuses on resource allocation in such a stochastic environment. For example, Smith [10] proposed a job scheduling algorithm for a single machine. However, they assumed that the mechanism is centralized with complete information on all the jobs, their significance, and processing time. Anderson and Potts [11] extended it to a scenario where the algorithm does not have access to complete knowledge of all the jobs. In all the above job scheduling algorithms, the designer of the algorithm makes implicit assumptions that the participating agents will act as instructed. Nisan and Ronen [12] proposed a job-scheduling algorithm that accounts for the strategic behavior of the participants. Heydenreich et al. [13] extended this idea to a strategic setting where the participants may manipulate the job processing time, arrival time of job, and the cost of waiting time. However the resulting mechanism is not decentralized and the equilibrium of the game is a myopic best response based (a weaker condition than Dominant strategy equilibrium). Christodoulou et al. [14] extended the LP-relaxation job scheduling problem into a mechanism design framework to also account for the strategic behavior of the participants. Jain et al. [15] developed an algorithm for allocating jobs in the cloud which is truthful-in-expectation which is primarily suited for cloud computing applications. Andelman et al. [16] showed a Fully Polynomial Time Approximation Scheme algorithm for scheduling jobs on a fixed number of machines that elicit truthful revelation with the goal of minimizing overall completion time.

In all of the job-scheduling algorithms the focus is only on optimizing some global objective function such as overall completion time or cost and they ignore individual objectives of the independent agents. Moreover, they do not have useful properties such as stability, individual rationality, and consistency.

The field of mechanism design focuses on designing mechanisms that have useful properties, but they lack studies on the optimal frequency of implementation of such mechanisms. In situations where agents repeatedly interact with one another, manipulations and strategic behavior are much more probable because of the knowledge about historic data. Therefore, there is a need to identify optimal frequency of implementation of mechanisms so that they satisfy useful properties and produces optimal matches when implemented in situations where agents repeatedly interact with one another over long periods of time.

To address this gap the central research question in this paper is: *what is the optimal period of matching for a given service arrival rates considering matching objectives such as average utility attained, number of successful matches and fairness in the distribution of utility?* We use simulation studies on a synthetic CBDM scenario to identify the optimal period of matching for various arrival rate of designers and perform Sobol sensitivity index [17] to study the robustness of the design period to the variabilities in the seeker arrival rate and availability of providers.

This paper is structured as follows. Section 2 describes the modeling of resource allocation in CBDM as a matching problem, Section 3 shows the results of simulation studies on a synthetic CBDM scenario and identifies optimal matching period, Section 4 studies the robustness of this optimal matching period and Section 5 presents the concluding remarks.

2 Modeling Resource Allocation in CBDM as a Stochastic Matching Problem

We model CBDM as a resource allocation problem, where service seekers (S) avail manufacturing resources from service providers (P). The sets of service seekers, and service providers are denoted by $S = \{s_1.s_2...., s_{|S|}\}$, and $P = \{p_1, p_2...., p_{|P|}\}$ respectively. Service seekers and providers will be collectively referred to as agents. The set of service seekers constitute the alternative set of service providers and vice-versa. Together, they form a bipartite set, and the resource allocation is formulated as a bipartite matching problem.

The first step in matching involves quantification of preferences of seekers and providers using the expected utility theory [18]. The next step is to generate the preference rank ordering of alternatives for each participant based on the expected utilities. The matching algorithm is then implemented to match the service seekers to the most suited service providers. This is called a single-period matching. These steps are elaborated further in Section 2.1.

In a stochastic environment, where the service seekers and providers arrive and exit the system as a continuous process over a long period of time (T) the matching mechanism needs to be implemented multiple times. The mechanism is implemented recursively after every fixed interval of time, t_{design}. The recursive implementation of the single-period matching after every fixed interval of time is called multi-period matching. The interval between two successive implementations of the matching mechanism is referred to as a matching cycle. During this period new service seekers place their service requests, the service providers complete their jobs assigned in previous cycle and become available. A suitable designed matching period t_{design} optimizes the outcome of the mechanism. Section 2.2 elaborates the modeling and implementation of multi-period matching in CBDM.

2.1 Single-period matching

Following the work of Fernandez et al. [19] we use a utility-based procedure to quantify the preference characteristics of the agents. The first step is to identify all the attributes that the agents consider while evaluating their alternatives. For example, all service seekers value certain attributes of the service providers such as the resolution of the machine they possess, the strength of the material offered etc. Similarly, service providers consider certain attributes such as printing time, size of the design while evaluating their choice set of alternatives. All these attributes are collectively referred to as $X = \{X_1, X_2, \ldots, X_n\}$.

Using standard utility assessment procedures [18] single attribute utility function f_{ki} can be obtained for any service seeker s_i towards attribute X_k. The single attribute utility functions are combined to obtain multi-attribute utility function $u_i(X) = u(f_{1i}(X_1), f_{2i}(X_2), \ldots, f_{ni}(X_n))$. $u_i(X)$ defines the preference characteristics of service seeker s_i towards an alternative characterized by attributes X. Assuming the additive form of the multi-attribute utility function, we have $u_i(X) = \sum_{k=1}^{n} w_{ki} f_{ki}(X_k)$ where w_{ki} is the weight that s_i associates to attribute X_k. The attributes that the service seeker do not care about are assigned a value of zero.

f_{ki} is the utility that the service seeker s_i attains for a certain value of the attribute X_k. In practice, X_k will not be a fixed value but would be a probability distribution over a range of possible values. For example, the service provider would offer a range of resolution for its printed parts based on the type of machine offered. $p_{kj}(X_k)$ denotes the probability density function (pdf) offered by service provider s_j over the range of possible values of attribute X_k.

The pdf of attributes offered by service provider p_j along with the multi-attribute utility function of service seeker s_i is used to calculate the expected utility that s_i attains after being matched to its alternative p_j. This is denoted by $E[u_{ij}(X)]$.

$$E[u_{ij}(X)] = \sum_{k=1}^{n} w_{ki} \int [f_{ki} p_{kj}] dx \quad (1)$$

Repeating the same steps for service providers, we obtain the expected utility service of provider p_j after being matched to service seeker s_i as

$$E[u_{ji}(X)] = \sum_{k=1}^{n} w_{kj} \int [f_{kj} p_{ki}] dx. \quad (2)$$

The utility attained by service seekers by being matched is also a non-increasing function of waiting time. The difference between the time period of each matching cycle (t) and mean arrival rate ($\frac{1}{\lambda}$) can be used as a nominal value for the waiting time of service seekers. The true utility that the agents gain after being matched is affected by the waiting time. In our model, we assume that the period of each matching cycle (t) is short enough that the temporal variability in utility can be neglected.

Based on the utility obtained for each agent being matched to their alternatives, the preference rank ordering of each agent over all their alternatives is generated. The preference rank ordering so generated is used in the DA algorithm 1 to match designers to manufacturers.

2.2 Multi-period Matching

The demand for service and availability of resources determine the market thickness in CBDM. Both arrivals of service requests and processing of services are modeled as stochastic processes. This section describes the modeling of these stochastic processes.

Set each $s_i \in S$ as unassigned and each $p_i \in P$ as totally unsubscribed
while *($\exists p_i \in P$ who is undersubscribed) and ($\exists s_k \in R_{p_i}$ not provisionally assigned to p_i)* **do**

1. s_i is first such s_k in R_{p_i} and s_i is provisionally assigned to p_j
2. unassign s_i from p_j and provisionally assign s_i to p_i
3. for each successor p_k on R_{s_i} remove p_k and s_i from each other's list

end

Algorithm 1: Single-period DA mechanism

2.2.1 Arrival of Service Seekers

The stochastic arrival of service seekers is modeled as a Poisson process with mean arrival rate λ. We chose a Poisson process to model the number of service seekers because (a) Poisson distribution models discrete events that occur in a finite and continuous interval of time, and (b) service seekers arrive from a wide range of sources independent of one another. The sources are designers or groups of designers trying to get their parts printed or manufactured. They are independent because the arrival time of a designer does not depend on the arrival time of other designers in the system. The set of service seekers on k^{th} matching cycle is denoted as S_k. The probability density function $\pi(|S_k|)$ over number of service seekers $|S_k|$ is given by Equation 3.

$$\pi(|S_k|) = \frac{e^{-\lambda}\lambda^{|S_k|}}{|S_k|!} \tag{3}$$

The mean arrival rate (λ) is a characteristic of the target population on which the mechanism will be implemented. The higher the demand for cloud-based services higher the value of (λ). The mean arrival rate varies from one setting to the next. The demand cannot be controlled by the mechanism designer. The goal is to design the matching period (t) for a given demand (or arrival rate λ) of the target population.

2.2.2 Availability of Service Providers

The set of service providers who are available on the k^{th} matching cycle is denoted by S_k; we have $S_k \subseteq S$. The availability of service provider p_j in a matching period depends on the complexity of the job assigned to them in the previous periods, the type of manufacturing resource they possess, and their average working time in a day (h_j). The complexity of service request (of the service seeker s_i that he/she got matched to) and the type of manufacturing resource (possessed by service provider p_j) determine the service time (τ_{ij}) needed for processing a request. For example, a service provider with a machine that uses a Stereolithography (SLA) process can print a design faster than a service provider possessing a machine that uses Fused Deposition Modeling (FDM) printing process. We assume that each service provider is available in a matching cycle if he/she completes the previous service request in the preceding matching cycles. $\frac{\tau_{ij}}{h_j t}$ is the number of matching cycles after which service seeker p_j is available after being matched to service seeker s_i. A service provider who is able to meet the service requirements faster is more available and therefore derives more utility during the matching duration T by being available on a higher number of matching cycles.

2.3 Matching Mechanism

Thekinen and Panchal [9] have shown that the Deferred Acceptance (DA) algorithm is the most suited matching mechanism in a totally decentralized design and manufacturing setting due to its properties such as stability, individual rationality, consistency, immunity to gaming behavior of the participating agents, monotonicity with respect to demand, and supply. Now, we extend this matching mechanism into a multi-period setting and determine the optimal period of implementation of the mechanism (t). The extension of the DA mechanism in a multi-period setting is described in Algorithm 2.

Set $avail_i = 1\ \forall\ \ p_i \in P$
for *($time \leftarrow t$ to T step t)* **do**
  if *time is t* **then**
    Set each $p_i \in P$ as unassigned;
  else
    Unassign those $p_i \in P$ who have $avail_i = 1$;
  end
  while *($\exists p_i \in P$ who is unassigned) and ($\exists s_k \in R_{p_i}$ not provisionally assigned to p_i)* **do**
    s_i is first such s_k in R_{p_i} and s_i is provisionally assigned to p_j;
    unassign s_i from p_j and provisionally assign s_i to p_i;
    for each successor p_k on R_{s_i} remove p_k and s_i from each other's list;
  end
  Set $avail_i = 0$ for all p_i in P who is assigned a seeker in this matching cycle;
  Set $avail_i = 1$ for those p_i who completed previous assignment;
end

Algorithm 2: Multi-period DA mechanism

3 Simulation Studies

We consider an illustrative scenario where 50 independent 3D-printer machine owners are offering manufacturing services. Service seekers are designers from a population who are trying to get their designs prototyped in the 3D printers. The service providers and service seekers are referred to as manufacturers and designers respectively. The arrival of designers is modeled as a Poisson process. The manufacturer p_j offer h_j working hours per day. Matching is done after every t days. We assume that each manufacturer can be matched to at most one designer in a matching cycle. This is not a limitation of the model but has been assumed for analysis purposes.

3.1 Data Collection

The manufacturers' attributes considered are machine volume, machine resolution (Res), the tensile strength (TS) of the material offered, manufacturer proximity whereas designer attributes were printing time, material requirement, and design dimensions (Vol). To generate the attributes of the designers, 100 different designs are downloaded from Thingiverse [20] and their characteristics such as design dimensions, printing time required in different 3D printers are recorded. Some of the sample designs used are shown in Figure 1. Attributes of a large sample size of designers are generated from these recorded attributes. Manufacturer data concerning the machine attributes are collected from the Senvol [21] database. The machine search mode on the database is used for searching machine features. Material properties of the material used in these 3D printers are collected from iMaterialise [22]. 50 unique material machine combinations are used to define the attributes of 50 manufacturers.

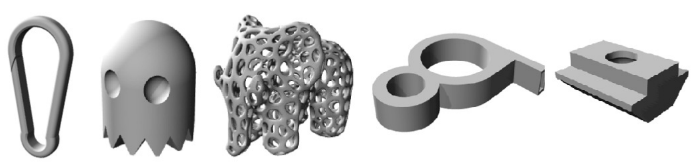

FIGURE 1. Samples of designs used in the simulation studies

TABLE 1. Range of values used for the attributes in the simulation studies

Attribute	Area (in^2)	Vol (in^3)	Res (mm)	TS (MPa)
min	3.4	4.5	0.01	14
max	279.3	67875.4	1	1800

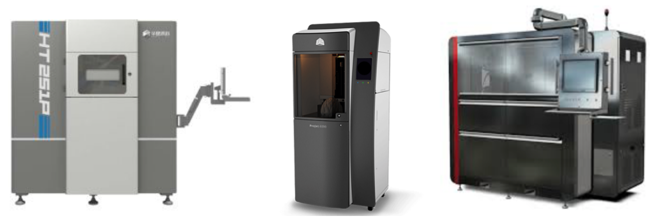

FIGURE 2. Examples of some of the 3D printers used in the simulation studies

3.2 Setting Parameters of the Simulation Studies

The arrival rate of designers is varied from 1 arrival per day to 50 arrivals per day. The efficiency of the mechanism is assessed over a duration of 30 days ($T = 30$). The mean printing time of the selected designs on the selected machines for simulation studies is approximately 5.99 hours. The working time (h) denoting the availability of the manufacturers on a day was set to low (1.5 hours per day) and high setting (24 hours per day). When $h = 24$ hours nearly all the manufacturers (including the matched ones) are available in every cycle of matching when period (t) is varied from 0.5 days to 10 days. However, when $h = 1.5$ hours some of the matched manufacturers will not be available on a few subsequent cycles after getting matched, particularly when $t < 4$ days.

3.3 Results of Simulation Experiments

The following measures were used to compare the efficiency at different matching period: a) total expected utility attained by the set of manufacturers (denoted by $\overline{EU_P}$), b) total expected utility attained by the set of designers (denoted by $\overline{EU_S}$), c) fraction of successful matches, and d) variation in utility distribution among service providers (denoted by $\overline{\sigma_{EU_P}}$). Total expected utility attained by the set of manufacturers is

$$EU_P(t_k) = \sum_{j=1}^{|P_k|} E[u_{ji}]\mathbb{1}_{\mathbb{M}^{-1}} \;\; where, \begin{cases} \mathbb{1}_{\mathbb{M}^{-1}} = 1, & M_{p_j}^{-1}(t_k) \in S_k \\ \mathbb{1}_{\mathbb{M}^{-1}} = 0, & otherwise \end{cases} \quad (4)$$

The total expected utility attained by set of designers is

$$EU_S(t_k) = \sum_{i=1}^{|S_k|} E[u_{ij}]\mathbb{1}_{\mathbb{M}} \qquad where, \begin{cases} \mathbb{1}_{\mathbb{M}} = 1 & M_{s_i}(t_k) \in P_k \\ \mathbb{1}_{\mathbb{M}} = 0 & otherwise \end{cases} \quad (5)$$

The fraction of successful matches in k^{th} matching cycle is given by $\frac{|P_k|}{|P|}$. The degree of unfairness in a matching cycle with respect to service providers is measured by the standard deviation of the utility distribution among all the service providers participating in that cycle ($|P_k|$) is

$$\sigma_{EU_P}(t_k) = \sum_{j=1}^{|P_k|} \left(\left(E[u_{ji}] - \frac{EU_P(t_k)}{|P_k|} \right)^2 \mathbb{1}_{\mathbb{M}^{-1}} \right) \tag{6}$$

All these measures are obtained by averaging over the entire matching duration T as shown in Equation 7. In Equation 7, $O(t_i)$ indicates the value of measure O at matching cycle t_i and $\lfloor \frac{T}{t} \rfloor$ is the number of matching cycles. Here, $\lfloor\rfloor$ denotes floor function. O is one of either EU_P, EU_S, σ_{EU_P}, the fraction of successful matches depending on the objective of the mechanism we consider for efficiency.

$$\overline{O} = \sum_{i=1}^{\lfloor \frac{T}{t} \rfloor} O(t_i) \tag{7}$$

The results showing the effect of matching period on each of the four objectives are discussed in the rest of this section.

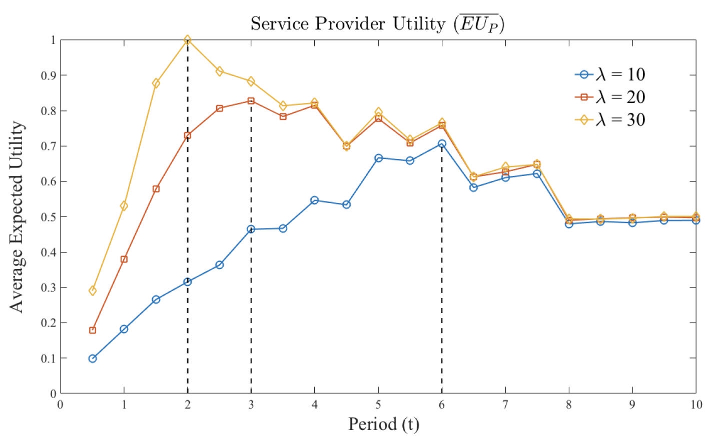

FIGURE 3. Total Expected Utility attained by the set of service providers as a function of matching period for various arrival rates

3.3.1 Service Provider Utility

When the period of matching cycle is increased, manufacturer (or service provider) utility ($\overline{EU_P}$) gets affected due to two reasons: a) the sample size of service seekers increases thereby increasing the number of alternatives to choose from, b) the number of matching cycles decreases over a fixed duration T thereby decreasing the average utility attained over an assessment duration T. Both of these causes have opposing effects on the average manufacturer utility.

When the matching period t is increased from 0, at low values of t, the effects due to a low sample size of service seeker alternative are more dominant considering the average utility attained by all service providers. This is because most of them remain unmatched in each matching cycle. Moreover, if the average service request processing time τ is lower, then the unfairness in utility distribution is more prominent at small values of matching period t. This is because the more sought-after manufacturer will be matched in each cycle while the others remain unmatched. If τ is high, then the less desirable manufacturers get matched due to the lack of availability of the manufacturers already matched in every matching cycle.

At large values of t, the marginal effect from an increased sample size due to increased t is less prominent. Now, the effect of a decreased number of matching cycles becomes more prominent. Therefore, as t is varied from 0 to large numbers there is an increase in service provider utility initially, followed by a decrease. The point where the effect of marginal increase in period on the service provider utility undergoes a transition from increase to decrease, is the optimal matching period.

From simulation studies, we obtain that the optimal matching period is $t_{design} = \frac{|P|}{\lambda}$. This is the period at which service providers attain the highest utility. This is not an exact point as the optimum shifts mildly due to stochasticity in arrival patterns, randomness in utility distribution, the working time of service providers. But over a wide range of arrival rate λ, t_{design} is the optimal matching period. The reason is that when $0 < t < t_{design}$ there are not enough service seekers in the system (in expectation) at the instance of matching to match all the service providers. Thus, when t is increased from 0 to t_{design} the effect of an increased number of service seekers is more pronounced than the effect of a decreased number of matching cycles. However, when $t > t_{design}$ there is sufficient number of service seekers to match all the service providers and only the quality of alternatives or number of matching outcomes to choose from improves. As a result, the effects due to a decreased number of matching cycles start taking more precedence, thereby decreasing the overall utility. Also, the presence of a $|P|$ number of service seekers does not guarantee that all service providers in P will get matched. This is because DA produces only stable matches and there may be no stable solution that gets all $|P|$ agents matched. In practice, the required number of service seekers is slightly higher than $|P|$ because of this restriction of stability. This is why in the simulation studies the actual optimal point is slightly higher than the design optimal point as seen in Figure 3. For example, when $\lambda = 1$ we have $t_{design} = 5$ but the actual optimum occurs at $t = 6$, and when $\lambda = 2$ we have $t_{design} = 2.5$ when the actual optimum is at 3. Thurber [23] showed that the number of stable matches under DA mechanism (denoted by $|M|$) when λt men and women are being matched is

$$|M| > \frac{1.509^{\lambda t}}{1+\sqrt{3}} \quad when \quad |\lambda t| \geq 1 \tag{8}$$

As a result, the actual optimum is not too far away from t_{design} as the number of stable matches grows in power of t.

3.3.2 Service Seeker Utility

The average expected utility per designer (or service seeker) decreases monotonically as the period of matching increases. This is because of the increased competition due to a higher number of designers using the same set of available resources. The average utility is not a good measure of matching efficiency as the total number of designs getting printed is not accounted for in this metric. Therefore, we use the total utility attained by all designers to quantify efficiency ($\overline{EU_S}$).

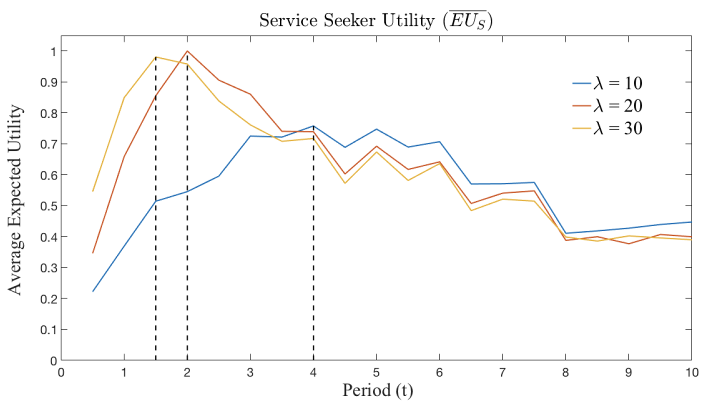

FIGURE 4. Total Expected Utility attained by the set of service seekers as a function of matching period for various arrival rates

Similar to service providers, there is an optimal period of matching where the effect due to a marginal increase in the number of designers becomes less prominent than the effect due to a decreased number of matching cycles. $\overline{EU_S}$ increases from $t = 0$ to this optimal point and decreases thereafter. From the results of simulation studies as shown in Figure 4, we conclude that the matching period that maximizes designer utility is $t_{design} = \frac{|P|}{\lambda}$. The actual optimum is slightly lower than the design optimal point. For example, when $\lambda = 1$ we have $t_{design} = 5$ but the actual optimum occurs at $t = 4$ and when $\lambda = 2$ we have $t_{design} = 2.5$ when the actual optimum is at 2. This is because some of the machines considered for the simulation studies were highly undesirable to prototype most of the designs and therefore there is not much increase in overall utility being matched to them. This is a very small fraction and the designed optimum is not too far off from the actual optimum.

3.3.3 Fraction of Successful Matches

From the simulation studies, the actual optimum period of designer and manufacturer was very close to the design period. In the case of manufacturers, the actual optimum was slightly higher than t_{design}, whereas it was slightly lower than t_{design} for designers. The efficiency lost because of this difference is the price paid for stability and randomness in the scenario such as variations in machine capacity. However, this loss is very small and the deviation of actual optimum from t_{design} is minor as seen from the simulation results. The number of successful matches is another important matching objective. In Figure 5 we show that the fraction of successful matches as a function of matching period. The steepest increase in the fraction of successful matches is seen from 0 to t_{design} as when $t < t_{design}$ there is not sufficient number of designers in the system to match all the manufacturers. When $t > t_{design}$ only the number of stable matching outcomes to choose from increases and the improvement in the fraction of successful matches with an increased period is small. Thus at $t = t_{design} = \frac{|P|}{\lambda}$ a 'nearly'optimal fraction of successful matches is achieved while optimizing the designer and manufacturer utility.

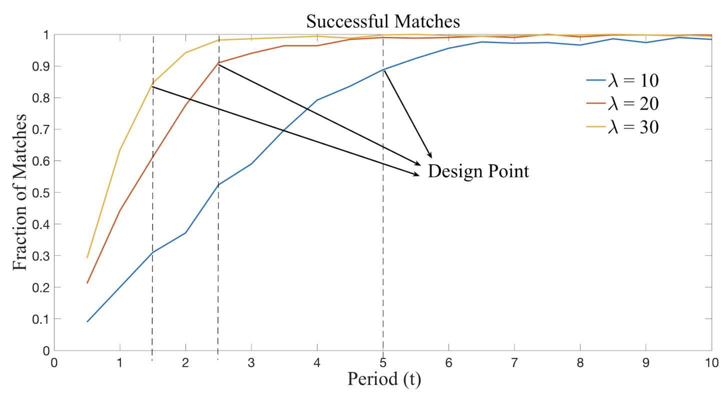

FIGURE 5. Fraction of successful matches as a function of matching period for various arrival rates

3.3.4 Fairness

From the perspective of the mechanism designer, the objective is not just to optimize for aggregate utility attained by the participating agents, but to also account for the fairness in the distribution of utility. When the period is very small ($t \to 0$) the matching outcome is unfair to the service providers. The most sought-after service provider keeps getting matched repeatedly. This is particularly true when $\frac{h}{\overline{\tau}}$ is large, i.e., when availability (or working time h) is high compared to the average service time ($\overline{\tau}$) as the most sought-after manufacturer is available in most of the matching cycles thereby gaining an advantage over the rest. This causes the unfair region of outcomes (or peak of the standard deviation of utility) to shift further away from design point (t_{design}), as seen in Figure 6. A working time of 24 hours per day for a mean printing time of 5.99 hours was used to study the effect of period on fairness in Figure 6.

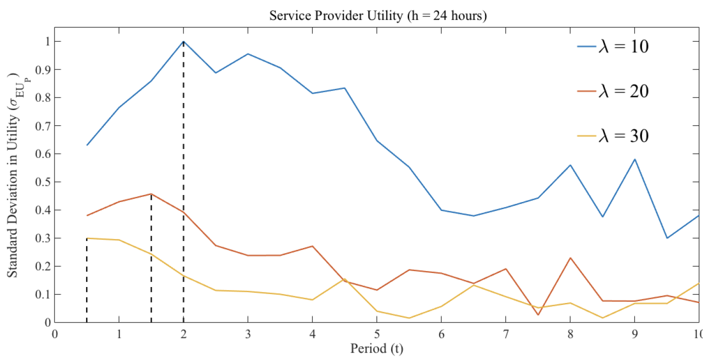

FIGURE 6. Standard deviation of Expected Utility attained by the set of service providers as a function of matching period for various arrival rates (when working time is $h = 24$)

The process was repeated for $h = 1.5$ hours per day and the unfair region shifted to higher values of matching periods, as shown in Figure 7. This is because of low working times, the matched manufacturers are not available in the subsequent matching cycles thereby allowing other manufacturers to get matched as well increasing the fairness.

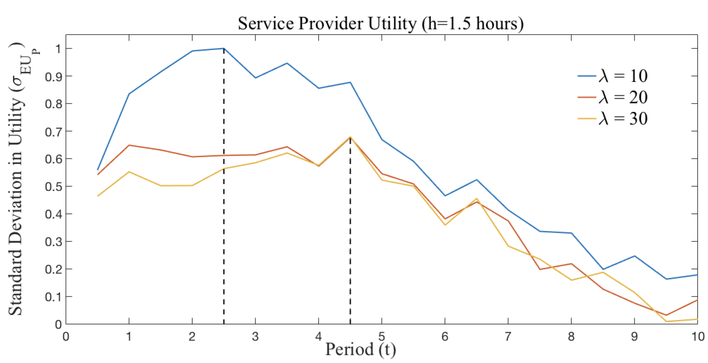

FIGURE 7. Standard deviation of Expected Utility attained by the set of service providers as a function of matching period for various arrival rates (when working time is $h = 1.5 hours$)

4 Robustness of the Optimal Period

From the simulation studies, an optimal period (t_{design}) was proposed for a given arrival rate. Now, the arrival rate may not remain constant over the entire matching duration T and is subjected to variabilities. For example, depending on the day of the week or time of the day, the arrival rate may differ from its nominal value. The optimal period was designed for the nominal value of arrival rate. It is important to measure the robustness of the optimal solution to such variabilities. We use sensitivity analysis to measure the robustness. Sensitivity analysis is an important tool in the validation and measure of robustness of the model [24]. Section 4.1 presents the framework used to perform sensitivity analysis and Section 4.2 discusses the results obtained from the analysis.

4.1 Framework for Sensitivity Analysis

Deferred Acceptance mechanism always picks the same set of stable matched pairs for an input set of agents and preference lists. The matching mechanism is a deterministic model [25] and uncertainty in output is only due to uncertainty in model parameters.

The inputs to the mechanism are preference distribution of both agents, availability of service providers (working time), number of service providers, and arrival rate of service seekers. Based on the input, the optimal matching period parameter (t_{design}) of the mechanism is designed. This depends only on the arrival rate. We study the robustness of various objectives to variabilities in arrival rate (λ) and matching period (t). The robustness of the following four objectives (O) are studied: a) utility attained by service providers, b) utility attained by seekers, c) fraction of successful matches, and d) fairness of matched outcome. These objectives were described in Section 3.3.

There are various metrics to quantify sensitivity; some of the examples are differential sensitivity [26], partial rank correlation coefficients [27], frequency domain approach [28]. However, we use Sobol sensitivity index [17] as our model is computationally inexpensive and this approach measures both linear and non-linear effects. Global sensitivity of input X_i on the objective O considering only linear effects is denoted by L_i and is given by Equation 9. Global sensitivity of input X_i on objective O considering both linear and non-linear effects due to interaction with other input variables is denoted by T_i and is given by Equation 10. Results of the sensitivity analysis are presented in Section 4.2.

$$L_i = \frac{Var_{X_i}(E_{X_{\sim i}}(O|X_i))}{Var_O} \tag{9}$$

$$T_i = \frac{E_{X_{\sim i}}(Var_{X_i}(Y|X_{\sim i}))}{Var_O} \tag{10}$$

4.2 Results of Sensitivity Analysis

The global sensitivity results for various objective functions are shown in Table 2. These results were verified using convergence studies. Convergence study for one of the matching objectives is shown in Figure 8. From Table 2 we observe that considering utility attained by seekers and providers as the matching objective, global sensitivity towards variability in arrival rate was much lower than sensitivity towards variability in the period. Thus the design point is robust towards variability in arrival rate when utility attained by the participating agents is the matching objective. While considering the sensitivity of fraction of successful matches and fairness in the distribution of utility both arrival rate and period have equal contribution. This is because these objectives are equally affected by variabilities in both the inputs. For example, increasing either the arrival rate or the matching period from 0 to very high values increases the fraction of successful matches monotonically from 0 to 1 as both results in an increased number of designer choice to choose from in the bipartite matching.

TABLE 2. Linear and total global sensitivity of various matching objective towards its inputs

Objective \ Variable	Arrival Rate (λ)		Period of cycle (t)	
	Linear	Total	Linear	Total
Utility of P ($\overline{EU_P}$)	0.176	0.454	0.571	0.849
Utility of S ($\overline{EU_S}$)	0.020	0.322	0.703	1.005
fraction of matches	0.320	0.687	0.338	0.705
$\overline{\sigma_{EU_P}}$ when $h = 24$	0.300	0.686	0.339	0.725
$\overline{\sigma_{EU_P}}$ when $h = 1.5$	0.004	0.100	0.925	1.021

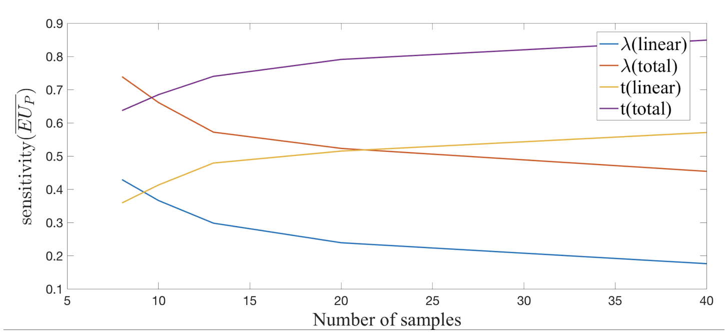

FIGURE 8. Convergence study for sensitivity. Sensitivity of total utility of service providers is plotted.

In Table 3 we compare the local sensitivity of various matching objectives towards arrival rate to the global sensitivity. Local sensitivity when $\lambda \in [100, 200]$ is much lower than global sensitivity. This is because for the range of matching period ($0.5 \leq t \leq 10$) considered, increasing the arrival rate from 100 to 200 only increases the quality of matches and not the number of successful matches. Thus most of the contribution to global sensitivity comes in the range $\lambda \in [0, 50]$ and that is also evident from the local sensitivity values in this range being higher than the global sensitivity for all the matching objectives considered.

TABLE 3. Local sensitivity of matching objectives towards arrival rate λ

Objective \ Range	Global	$[0, 50]$	$[100, 200]$
Utility of P ($\overline{EU_P}$)	0.454	0.645	0.046
Utility of S ($\overline{EU_S}$)	0.322	0.514	0.040
fraction of matches	0.687	0.466	0.287
$\overline{\sigma_{EU_P}}$ when $h = 24$	0.686	0.752	0.207
$\overline{\sigma_{EU_P}}$ when $h = 1.5$	0.100	0.219	0.035

5 Conclusion

From the simulation studies, we propose that in a multi-period matching with seekers arriving from the population of a wide range of independent sources and a fixed number of independent providers offering manufacturing resource, the optimal matching mechanism is Deferred Acceptance mechanism with providers as proposers implemented at an interval of $t_{design} = \frac{|P|}{\lambda}$. This is because when $t < t_{design}$ the effect of an increased number of service seekers is more pronounced than the effect of decreased number of matching cycles, but after $t > t_{design}$ the increased number of service seekers only increases the quality of alternatives to choose from and does not influence the number of successful matches. Optimality is considering the number of successful matches, utility attained by both service seekers and service providers as the criteria. When the matching objective is the number of successful matches then t_{design} gives an approximately optimal solution. This is because $|P| = \lambda t_{design}$ does not guarantee everyone to be matched as there may be no such stable solution. This loss in efficiency is the price paid for stability. Similarly, the actual optimal point provider utility is at a period slightly higher than t_{design} and for seeker utility is at a period slightly lower than t_{design}. This happens due to lack of stable solutions that can match all the service providers. However, this difference is not significant as the number of stable solution grows as a power of matching period. Therefore, for design purposes t_{design} is the optimal period of matching that maximizes the matching objectives. The minor loss in efficiency is the price paid for stability. The designed period works well even from fairness standpoint when $\frac{\overline{\tau}}{h}$ is small compared to t_{design}. A low ratio $\frac{\overline{\tau}}{h}$ corresponds to a situation of high service rate as the providers are offering relatively higher number of working hours (h) for the same mean service processing time ($\overline{\tau}$).

Using sensitivity studies, the designed matching period was also found to be robust towards variabilities in arrival rate. Such a matching mechanism satisfies useful properties such as stability, consistency, immunity to strategic behavior, individual rationality, resource monotonicity, population monotonicity and at the same time optimize the matching objectives. An assumption made in these studies is that the temporal variation in utility is negligible. This is valid in CBDM applications where the matching period is small enough. However, in other applications such as kidney exchange, this cannot be neglected. When utility varies with waiting time, properties such as individual rationality, stability, consistency will be affected by matching period. For example, if prolonged for a long duration the matched outcome will not be individually rational as the agent would rather remain unmatched. In future studies, this assumption can be relaxed to study the interplay between satisfaction of properties of mechanism and efficiency.

Acknowledgments

The authors gratefully acknowledge financial support from the US National Science Foundation (NSF) through grants 1360361 and B.F.S. Schaefer award from the School of Mechanical Engineering at Purdue University.

REFERENCES

[1] Wu, D., Thames, J. L., Rosen, D. W., and Schaefer, D., 2012. "Towards a cloud-based design and manufacturing paradigm: looking backward, looking forward". In ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, pp. 315–328.

[2] Kesten, O., 2006. "On two competing mechanisms for priority-based allocation problems". *Journal of Economic Theory,* **127**(1), mar, pp. 155–171.

[3] Gale, D., and Shapley, L. S., 1962. "College admissions and the stability of marriage". *The American Mathematical Monthly,* **69**(1), jan, p. 9.

[4] Balinski, M., and Sönmez, T., 1999. "A tale of two mechanisms: Student placement". *Journal of Economic Theory,* **84**(1), jan, pp. 73–94.

[5] Ergin, H. I., 1999. "Consistency in house allocation problems". *Journal of Mathematical Economics,* **34**, pp. 77–97.

[6] Ehlers, L., Klaus, B., and Pápai, S., 2002. "Strategy-proofness and population-monotonicity for house allocation problems". *Journal of Mathematical Economics,* **38**(3), nov, pp. 329–339.

[7] Ehlers, L., and Klaus, B., 2004. "Resource-monotonicity for house allocation problems". *International Journal of Games Theory,* **32**(4), aug.

[8] Mandhan, N., J., T., Lo, A., and J.H.Wu, P., May 9-13, 2016. "Matching designers and 3d printing service providers using gale-shapley matching". In Tools and Methods of Competitive Engineering (TMCE 2016).

[9] Thekinen, J., and Panchal, J. H., 2017. "Resource allocation in cloud-based design and manufacturing: A mechanism design approach". *Journal of Manufacturing Systems,* **43**, apr, pp. 327–338.

[10] Smith, W. E., 1956. "Various optimizers for single-stage production". *Naval Research Logistics (NRL),* **3**(1-2), pp. 59–66.

[11] Anderson, E. J., and Potts, C. N., 2002. "On-line scheduling of a single machine to minimize total weighted completion time". In Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, pp. 548–557.

[12] Nisan, N., and Ronen, A., 2001. "Algorithmic mechanism design". *Games and Economic Behavior,* **35**(1-2), pp. 166–196.

[13] Heydenreich, B., Müller, R., and Uetz, M., 2010. "Mechanism design for decentralized online machine scheduling". *Operations research,* **58**(2), pp. 445–457.

[14] Christodoulou, G., Koutsoupias, E., and Kovács, A., 2007. "Mechanism design for fractional scheduling on unrelated machines". In International Colloquium on Automata, Languages, and Programming, Springer, pp. 40–52.

[15] Jain, N., Menache, I., Naor, J. S., and Yaniv, J., 2014. "A truthful mechanism for value-based scheduling in cloud computing". *Theory of Computing Systems,* **54**(3), pp. 388–406.

[16] Andelman, N., Azar, Y., and Sorani, M., 2005. "Truthful approximation mechanisms for scheduling selfish related machines". In Annual Symposium on Theoretical Aspects of Computer Science, Springer, pp. 69–82.

[17] Sobol, I. M., 2001. "Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates". *Mathematics and computers in simulation,* **55**(1-3), pp. 271–280.

[18] Keeney, R. L., and Raiffa, H., 1993. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press.

[19] Fernandez, M. G., Seepersad, C. C., Rosen, D. W., Allen, J. K., and Mistree, F., 2005. “Decision support in concurrent engineering - the utility-based selection decision support problem”. *Concurrent Engineering,* **13**(1), mar, pp. 13–27.

[20] , 2016. Thingiverse. http://www.thingiverse.com/. Accessed: 2016-03-27.

[21] , 2018. Senvol. http://senvol.com/database/. Accessed: 2018-03-07.

[22] iMaterialise, 2015. imaterialise. http://i.materialise.com/. Accessed: 2016-03-27.

[23] Thurber, E. G., 2002. “Concerning the maximum number of stable matchings in the stable marriage problem”. *Discrete Mathematics,* **248**(1-3), pp. 195–219.

[24] Irving, A., 1992. “Stochastic sensitivity analysis”. *Applied mathematical modelling,* **16**(1), pp. 3–15.

[25] Anderson, D. R., Sweeney, D. J., and Williams, T. A., 1996. *Statistics for business and economics*, Vol. 6. West Publishing Company St. Paul.

[26] Tomovic, R., and Vukobratovic, M., 1972. “General sensitivity theory”.

[27] McKay, M. D., Conover, W., and Whiteman, D. E., 1976. Report on the application of statistical techniques to the analysis of computer codes. Tech. rep., Los Alamos Scientific Lab., N. Mex.(USA).

[28] Schruben, L. W., and Cogliano, V. J., 1981. “Simulation sensitivity analysis: A frequency domain approach”. In Proceedings of the 13th conference on Winter simulation-Volume 2, IEEE Press, pp. 455–459.

