Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

MATCHMAKING IN CROWD-SHIPPING PLATFORMS: THE EFFECTS OF MEDIATOR CONTROL

Preetam Kulkarni
Caroline C. Krejci

Department of Industrial, Manufacturing, and Systems Engineering
The University of Texas at Arlington
500 West First Street
Arlington, TX 76010, USA

ABSTRACT

A critical design decision for crowdsourcing platforms is the degree to which the platform mediator controls participant interactions. Platforms having a centralized model of mediation optimize for convenience, speed, and security in participant interactions, while platforms operating under decentralized control require greater user effort but offer them greater control and agency. The research described in this paper is a preliminary study using agent-based modeling to evaluate and compare the performance of crowd-shipping platforms with centralized/decentralized control over matchmaking of carriers and senders. Results indicate that centralized matchmaking protects the platform from premature failure when initial carrier/sender participation is low. Furthermore, when the platform's assignment algorithm is designed to maximize platform revenue, subject to meeting carriers' profit expectations, centralized matchmaking will tend to outperform decentralized matchmaking for both the mediator and the carriers.

1 INTRODUCTION & BACKGROUND

Crowdsourcing is the concept of seeking knowledge, goods, or services from a network of people (i.e., the crowd) to accomplish tasks that would otherwise require considerable resources or even be impossible for an individual or organization to undertake on their own. Owing to the sharing of the crowd's underutilized capacity in exchange for money, crowdsourcing is also referred to as the "sharing economy". Since first being defined nearly two decades ago (Howe 2006), crowdsourcing has grown by leaps and bounds (Wazny 2017), with the number of research publications on crowdsourcing and the sharing economy increasing to 900 by 2015 (Ghezzi et al. 2018).

Crowdsourcing relies on online platforms to facilitate user interactions and to collect, process, share, and monetize user data. The way in which these platforms operate not only impacts their growth and success but also influences people's lives and societal organization (Dijck et al. 2018). To gain a better understanding of how crowdsourcing platforms impact society, much of the research on the sharing economy has focused on platform design. Ideally, platform mediators design their platforms to facilitate participant interactions in a way that generates value for all stakeholders (Kohler 2015), thereby attracting a network of users that grows and sustains itself over time (Wirtz et al. 2019). By contrast, poor design can result in low user adoption and retention, leading to early platform failure (Evans and Schmalensee 2010). Furthermore, if a platform is designed to pursue economic growth at the expense of stakeholder wellbeing, it may generate value for the mediator, but it can have negative consequences for society (Calo and Rosenblat 2017).

One critical design decision for crowdsourcing platforms is the degree to which the platform mediator controls participant interactions. On one end of the spectrum, platforms having a *centralized* model of mediation offer participants full automation with respect to key platform functions, including matchmaking,

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

pricing, and payment. These platforms optimize for convenience, speed, and security in participant interactions. In return, the mediator acts as an “authoritative presence” that governs participants’ choices and interactions, as well as harvesting their data (Sutherland and Jarrahi 2018). Most well-known crowdsourcing platforms, including Uber and Airbnb, operate via centralized control. On the other end of the spectrum, platforms with a *decentralized* model of mediation tend to exert minimal control over participant exchanges. Participants have greater responsibility for performing platform tasks but maintain greater control and agency as a result. Such platforms, which include Freegle and Couchsurfing, leverage the resources and innovation of their users and rely on self-organization (Gerber 2021). The degree of control retained by the mediator can vary across different aspects of a platform. For example, it could offer automated and secure financial transactions (centralized control) but require participants to perform their own matchmaking (decentralized control).

Determining the appropriate degree of platform mediator control over matchmaking is a particularly important design decision. Matchmaking involves connecting participants across the platform’s network based on what they are looking for and what they can provide. Matching occurs through two methods: *algorithmic assignment*, in which an algorithm assigns participants to one another (centralized control), or *active searching and sorting*, in which participants evaluate and select their peers (decentralized control). With algorithmic assignment, matching of supply and demand is optimized according to users’ profile data and ratings, as well as real-time data such as participant locations, service requests, and availability (Möhlmann et al. 2021). Algorithms may also employ mechanisms like surge pricing to facilitate matching by influencing and incentivizing participant interactions (Wirtz et al. 2019). Automated matching is convenient for users, allows them to connect with other users in real time, and can benefit society by connecting community members according to their interests or needs (Carroll and Bellotti 2015). However, the logic underpinning matching algorithms may be entirely opaque to participants and can therefore subvert their autonomy (Möhlmann and Zalmanson 2017) or even run counter to their own interests, thereby encouraging some users to circumvent the platform to regain some control (Lee et al. 2015).

By contrast, with active searching and sorting, users find and evaluate potential matches on their own, using information available via the platform, such as profile information and user reviews, or by communicating directly with other users. Decentralized control over matchmaking gives participants greater control over the process, allowing for greater democratization of the sharing economy (Scholz 2016) and enabling participants to efficiently self-organize (Gerber 2021). Furthermore, by relying on user contributions rather than mediator-created algorithms, decentralized platforms can cost less to create and maintain, allowing them to charge users lower fees (Frenken 2017). Cost-efficient platforms can also be leveraged in the public sector to rapidly expand capacity to serve rare peaks in demand (e.g., during large-scale events or disasters) with less public investment (Frenken 2017). However, decentralized matchmaking requires effort from participants, and the investment of time and energy into establishing relationships and trust with other participants can be prohibitive (Lampinen et al. 2015). Research has also shown that relying on profile information to inform matchmaking decisions can result in racial and socioeconomic discrimination (Schor and Attwood-Charles 2017). Additionally, emergent user self-organization is unpredictable and may lead to unexpected and undesirable system behavior (Gerber 2021).

Most existing research on platform design has focused on well-known platforms that operate via centralized mediation (Ntouros et al. 2021). However, centralized platforms do not represent the community-oriented aspects of the sharing economy, nor do they accurately embody the concept of bottom-up organizing, since the platform manages user interactions (Gerber 2021). There is a need for research that takes a broader human-centered view of optimization in the sharing economy (Dillahunt et al. 2017), and decentralized mediation may offer a better path forward in this regard (Light and Miskelly 2015). Thus, the design of decentralized platforms is a promising area for future research (Sutherland and Jarrahi 2018).

The research described in this paper is a preliminary study to evaluate and compare the performance of crowdsourcing platforms with centralized/decentralized control over matchmaking. Agent-based modeling is used to simulate the behavior of an artificial crowd-shipping platform with different levels of mediator control over matchmaking. Crowd-shipping (also referred to as “crowd logistics”) is the outsourcing of transportation and delivery of parcels or freight to occasional carriers drawn from a crowd of public and private travelers. Crowd-shipping services are coordinated by a platform that connects participants in a two-sided market: “senders”, who want to ship items, and “carriers”, who are willing to complete the delivery in exchange for money (Punel and Stathopoulos 2017). By leveraging the crowd’s excess resource capacity, senders gain temporary access to asset (i.e., vehicle) ownership benefits at a reduced cost and receive service that can be faster, more efficient, and more reliable than conventional delivery and courier companies (Castillo et al. 2018). Despite its many potential benefits, however, most crowd-shipping initiatives fail to attract a critical mass of participants to provide reliable and convenient service (Rougès and Montreuil 2014).

The most commonly used method for designing matchmaking logic in crowd-shipping platforms is optimization via mathematical modeling. Most models assume only the mediator’s point of view, with an objective(s) of minimizing total distance traveled, maximizing service levels, and/or increasing load utilization (Cleophas et al. 2019). For example, Le et al. (2021) used linear programming to evaluate how different pricing schemes impact optimal routing and matching of senders and carriers. Nieto-Isaza et al. (2022) used Bender’s decomposition algorithm to solve a two-stage stochastic transportation network design problem for a crowd-sourced last mile delivery system where the first stage of the decision was to locate mini-depots which allowed flexibility in matching demand to supply and the second stage allocated crowd carriers to specify delivery requests. A few models capture users’ requirements as well, such as ensuring a fair allocation of workload across all carriers (Chen et al. 2014) and allowing carriers to reject assignments that do not meet their criteria (Gdowska et al. 2018).

By contrast, agent-based modeling (ABM) is well-suited to modeling the heterogeneous preferences and decisions of autonomous senders and carriers, as well as their interactions. For example, Devari et al. (2017) used ABM to design a crowd-shipping system in which the carrier-sender network is derived from participants’ social networks. Westelaken and Zhang (2018) developed an ABM of a crowd-shipping platform to study how variables such as detour distance, carriers’ degree of flexibility, and the amount that carriers are paid for deliveries impact their decisions to participate in the platform. Chen and Chankov (2017) created an ABM to determine how carriers’ flexibility in accepting delivery requests affects platform service levels. Mittal et al. (2021) used ABM to study the impacts of different levels of initial sender/carrier participation on a crowd-shipping platform’s performance over time. However, none of these models has been used to study the effects of different levels of platform control over matchmaking.

In the following sections, the crowd-shipping ABM is described in detail, and a set of experiments to test the impact of different degrees of platform mediator control over matchmaking is proposed. The results of these experiments are presented, and the implications of these results for platform design are discussed.

2 MODEL DESCRIPTION

The ABM has been developed using NetLogo 6.2.2 and is described using Overview, Design Concepts and Details (ODD) protocol (Grimm et al. 2010).

2.1 Purpose

This research describes an abstract agent-based model that is used to simulate a crowd-shipping network and understand the performance of the platform over time, given different levels of platform control over matchmaking and different assumptions about initial carrier and sender participation. Platform performance is captured with respect to its ability to survive initial launch, the number of senders and carriers participating in the platform over time, cumulative platform revenue, average carrier profit per delivery, and average sender savings per delivery.

2.2 Entities, State Variables and Scales

Senders, carriers, and destinations are the three agent types. There are 250 potential senders and 250 potential carriers, and it is assumed that agents are either one or the other and are not allowed to switch roles. Each time-step (tick) represents one day. There are a total of eight randomly-located delivery destinations whose location changes with every simulation run. Each day, senders randomly select 1-2 destinations and submit delivery requests to the platform.

Attributes of sender and carrier agents include:

- Participation status of an agent on a given day.
- List of number of delivery requests fulfilled (senders) or number of deliveries completed (carriers) in each tick.
- Length of the list of IDs of agents who are "friends"; distributed N ~ (7,2).
- Number of days after which a sender or a carrier agent evaluates its participation decision; distributed N ~ (5,2).
- Average number of matches per day needed for the agent to continue participation or to join platform (assumed to be one match per day).

Sender-specific attributes include:

- Number of pickup requests per day; distributed U ~ (1,2).
- List of carrier IDs who matched with them in a given day.
- Value of time spent on delivering a package (assumed to be 17% of the trip cost).

Carrier-specific attributes include:

- Home location coordinates.
- List of sender IDs that matched with the carrier in a given day.
- List of destinations visited by the carrier in a given day.
- List of distance traveled for each trip in the current day.
- Limit on number of trips that a carrier is willing to complete each day.
- Minimum required profit per day, as per the distribution described by Le et al. (2019).
- Limit on the distance a carrier is willing to travel on a single trip (Le et al. (2019)).

2.3 Process Overview and Scheduling

At the beginning of each tick, agents whose evaluation periods have ended will evaluate their current platform participation. Agents that are currently participating will choose to discontinue if they have been unable to find a at least one successful match per day on average over the evaluation period. Similarly, an agent that is not currently participating would decide to join the platform if its friends have been able to find one match on average. Senders then submit their delivery requests (i.e., destinations) for the day to the platform, and carriers either select the requests they wish to fulfill, according to their level of profitability (decentralized control), or they are assigned to requests by the platform (centralized control). The carriers complete their chosen/assigned deliveries, and all participants update their participation history (i.e., the number of successful matches they experienced).

2.4 Design Concepts

2.4.1 Basic Principles

A carrier will match with a sender if the delivery request does not exceed the carrier's distance tolerance and the sender's proposed payment amount meets the carrier's minimum expectations for profit. These values were determined according to empirical research done by Le et al. (2019), who reported 65% of people are willing to travel at most 10 miles to deliver a package and the remaining 35% have a distance tolerance ranging from 20 miles to 50 miles.

2.4.2 Emergence

The number of participants and the platform revenue may increase or decrease as time progresses and this depends on the interaction of the agents and the satisfaction levels of the agents as determined by their decision-making criteria. Carriers who are not participating are motivated to join if they see that enough senders who they can match with are on the platform and vice versa. This phenomenon is related to network effects and is closely related to having a critical mass of participants in the initial days of the platform so that it sustains and grows in the future (Gruenbaum 2015). Hence the platform performance metrics being considered in this research are all emergent properties because they are dependent on multiple interrelated causal relationships.

2.4.3 Adaptation

Participating senders and carriers keep track of their number of successful matches each day, and if they notice that they are not able to find enough matches, they leave the platform.

2.4.4 Objectives

Senders try to save time and money by outsourcing the task of delivering packages to potential carriers who try to make a profit by delivering packages on the sender's behalf. If senders make the delivery by themself, then their distance traveled is twice the distance from sender location to the destination. If a carrier makes a delivery for a sender, the distance traveled by the carrier is equal to the sum of the distance from its current location to the sender's location and the distance from the sender's location to the destination. Senders and carriers both consider the basic cost of delivery to be a dollar for each unit distance traveled. Senders consider their total cost of a trip to be the sum of the basic travel cost and their value of time which is 17% of the basic travel cost. However, carriers each have a minimum pre-determined profit per trip as described by (Le et al. 2019) and this cost is added to their basic travel cost. The platform charges 5% of a carrier's trip cost as its revenue. The sum of platform's revenue and the carrier's trip cost is the delivery cost. The delivery cost is compared with a sender's trip cost, and if the delivery cost is less than or equal to the sender's cost, the sender and carrier are matched. Platform mediators try to increase platform revenue by helping carriers find matches.

2.4.5 Interaction

Participating agents interact with potential participants of their own type by sharing information about the number of matches they were able to find in the current day. If potential participants observe that the participating agents in their social network have been able to find one match on average, they decide to join the platform.

2.4.6 Stochasticity

Destinations are randomly positioned at the start of each simulation run. For the senders, number of pickup requests can either be 1 or 2 and this value is uniformly distributed. The carriers have a different distance tolerance and profit expectation based on a distribution discussed by Le et al. (2019), however, their daily trip limit is normally distributed. Number of friends and the evaluation period is normally distributed for both the carriers and senders.

2.4.7 Observation

Platform performance metrics are collected in each tick and include daily platform revenue, cumulative platform revenue, and number of successful deliveries. At the individual agent level, carrier/sender participation, utilization, and average profit/savings per tick are collected.

2.5 Initialization

At the beginning, 500 agents are created and half of them are senders and the remaining half are carriers. Eight destinations are created at random locations during each initialization; however, they remain at the same location throughout the simulation run. Every agent is assigned initial attribute values. Then all the agents build their social networks by randomly choosing friends of the same kind i.e., senders choose senders as friends and carriers choose carriers as friends. All agents initially set their participation status to "No"; however, depending on the experimental setting, a predetermined number of randomly-selected senders and carriers are chosen to participate initially and these agents change their participation status to "Yes".

2.6 Input Data

No input data is used to represent time-varying processes (Grimm et al. 2010).

2.7 Sub-models

2.7.1 Sub-model 1 – Join or Leave the Platform

At the beginning of each tick, an agent evaluates whether to leave or join the platform if the number of ticks since its last evaluation is equal to its participation evaluation period. A participating agent would leave the platform if the agent, on average, is not matched at least once per tick during the evaluation period. On the other hand, if an agent is currently not participating, the agent will start participating if all of its friends were able to find at least one match on average in the current tick.

2.7.2 Sub-model 2 – Matchmaking

Depending on the experimental scenario, one of three different modes of matching carriers and senders is employed. In the decentralized matchmaking mode, there is no intervention by the platform in the matching process. A carrier is randomly chosen from the unassigned carriers list to try to match with a randomly chosen unassigned sender's delivery request. The carrier will continue to seek a match until it finds one that satisfies its profit criterion. If the carrier is able to match with the sender, a new carrier is selected and the process continues as discussed before. If the carrier is unable to match with any sender, the carrier is removed from the unassigned carriers list. To ensure that a participating carrier that reaches its trip limit or a sender that finds a match for all of its destinations is not selected again for matching by the algorithm, the respective agent is removed from the list of unassigned carriers or unassigned senders.

In the modified decentralized matchmaking mode, the carrier is still responsible for evaluating and selecting a delivery request. However, the platform provides some guidance by restricting the carrier's awareness of available senders to a specific radius from its current location. By doing this, each carrier only tries to match with a sender whose location is within that radius. The rest of the matchmaking process is similar to that of the decentralized process described above.

In the centralized matchmaking mode, the unassigned carriers and unassigned senders lists are used to create a list of all possible carrier-sender-destination matches. The elements of this list are then used as variables to solve a linear programming problem where the objective is to maximize the platform revenue. As the carrier changes its position after each delivery, a new problem needs to be solved to look for other possible matches and hence, in a given tick, the process of solving the problem continues until the solver fails to find any feasible solution, as shown in the algorithm below.

Algorithm: Centralized Matchmaking

```
Input: Unassigned carriers and Unassigned senders lists
Output: Matched sender-carriers and completed trips
Initializations: "Result" as an empty list
While (length of "Result" $\neq 0$)
    Create list of possible carrier-sender-destination matches (CSD)
    Write and solve a linear program as follows:
        Maximize Platform revenue
        Subject to:
            Carrier trip cost $\leq$ Sender cost for all matches in CSD
            Carrier trip distance $\leq$ Carrier Distance tolerance for all matches in CSD
            Each carrier is assigned to only one sender
            Each sender-destination is assigned to only one carrier
            Non-negativity constraints
    "Result" is updated based on the solution of the linear program
    Carriers are matched to senders based on the "Result"
    Carriers execute trips assigned to them and stay at the destination location
    Remove carriers who reached their trip limit from "Unassigned carriers" list
    Carriers who reached their trip limit are sent back to their home location
    Remove senders who have zero pickup requests from "Unassigned senders" list
end
```

2.7.3 Update Participation History

At the end of each tick, the participation history for each carrier and sender is updated: senders record the number of successful deliveries they experienced in the current tick and carriers record the number of deliveries made in the current tick.

3 EXPERIMENTATION AND RESULTS

To understand the impact of initial carrier and sender participation on the platform's performance, three levels of initial participation were tested: 7, 18 and 30. This yielded nine possible scenarios, represented in the figures below by "(C, S)" where 'C' is the number of initial carriers and 'S' is the initial number of senders. These nine experimental scenarios were run for all the three levels of platform control over matchmaking for a total of 27 scenarios. Each of the 27 scenarios was run for 60 ticks for 30 replications. The run length was set to 60 ticks based on the observation that by the 50th tick, all performance metric values had stabilized.

Figure 1 shows the number of replications for each scenario in which the platform experienced sufficient participation from both carriers and senders to keep the platform operational for 60 ticks (i.e., participation did not drop to zero). Results indicate that having enough participants initially (ideally 18 carriers and senders) is crucial for the success of a platform, regardless of the mode of matchmaking algorithm being used. It is interesting to note that having fewer initial carriers than initial senders reduces the platform's chances of success compared to the opposite situation because carriers cannot keep up with the number of delivery requests if they are fewer in number compared to senders. Hence, dissatisfied senders leave the platform and carriers will not find enough matches and they also leave the platform. However, the results suggest that centralized matchmaking somewhat protects a platform from early failure. For example, when there are 7 carriers and senders initially, the number of successful simulations for the decentralized algorithm was lowest at 13, while the modified-decentralized algorithm increased this number to 18. This is because the platform helps carriers by restricting their awareness of senders to a certain radius and this make it easier for them to find a match. However, centralized matchmaking outperformed the others by achieving 23 successful replications.

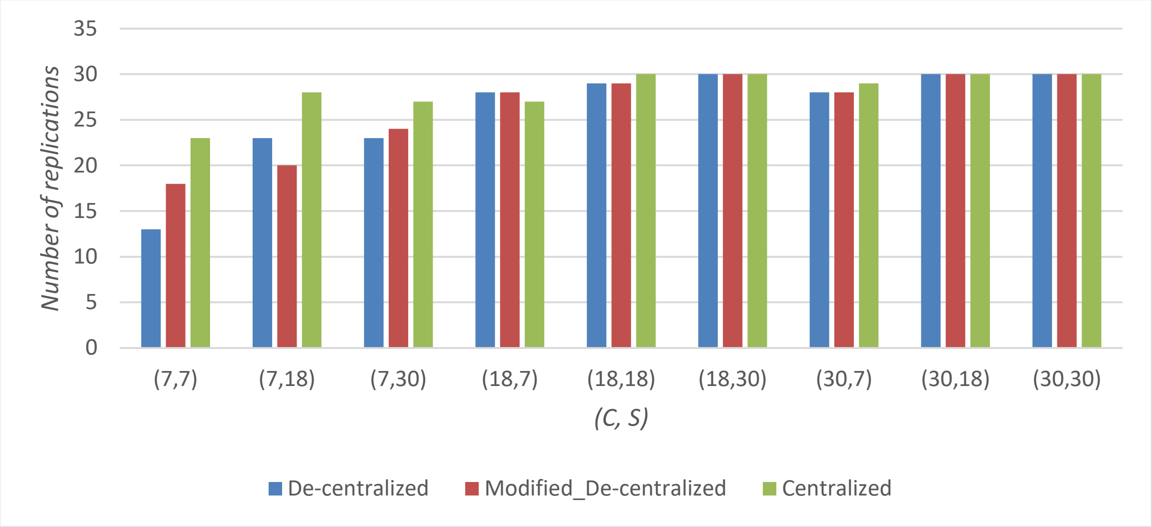

Figure 1: Number of successful replications in all scenarios.

To ensure equivalent comparison of the platform performance metrics, for the remaining analyses, data from unsuccessful replications (i.e., in which the platform failed before the 60th tick) were excluded. For successful replications, the average participation in the 60th tick for decentralized and modified-decentralized matchmaking was 170 carriers and 180 senders, while centralized matchmaking yielded a final participation of 180 carriers and 200 senders, on average (where the maximum possible participation is 250 senders and 250 carriers).

The average cumulative platform revenue for each experimental scenario is shown in Figure 2. The centralized matchmaking model always provides the highest revenue for the platform, which is unsurprising because the objective of the matchmaking algorithm is to maximize platform revenue. The modified-decentralized mode of matchmaking underperforms compared to decentralized matchmaking in most cases because the limited search radius reduces carriers' ability to find a good match. It can also be noticed that the variation in the platform revenue decreases as the number of initial participants increase, which emphasizes the need to have enough initial participants.

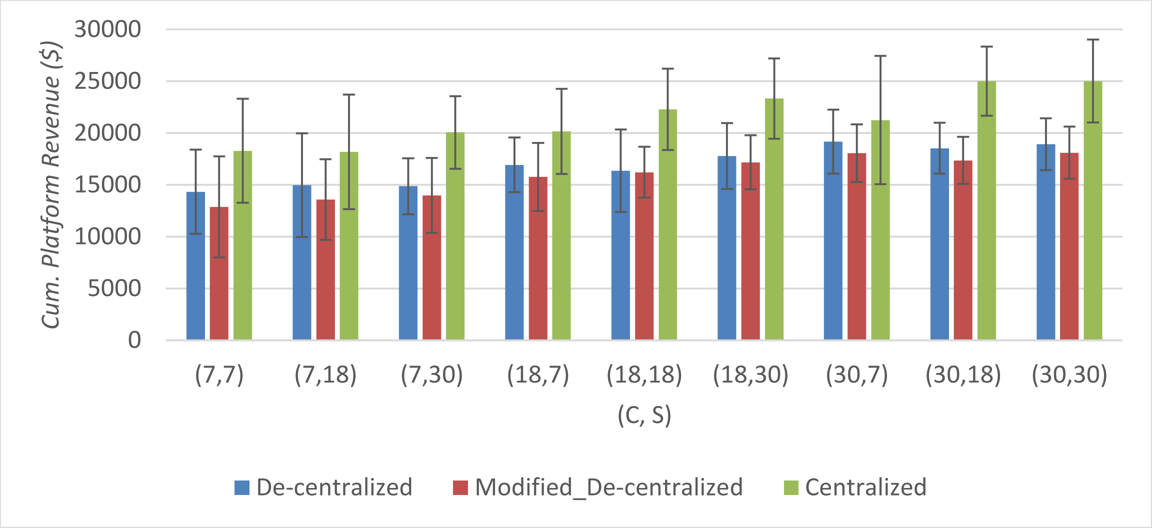

Figure 2: Average cumulative revenue of the platform per simulation in all scenarios.

The results in Figure 3 indicate that carriers earn similar average profits regardless of the mode of matchmaking or the initial sender/carrier participation values.

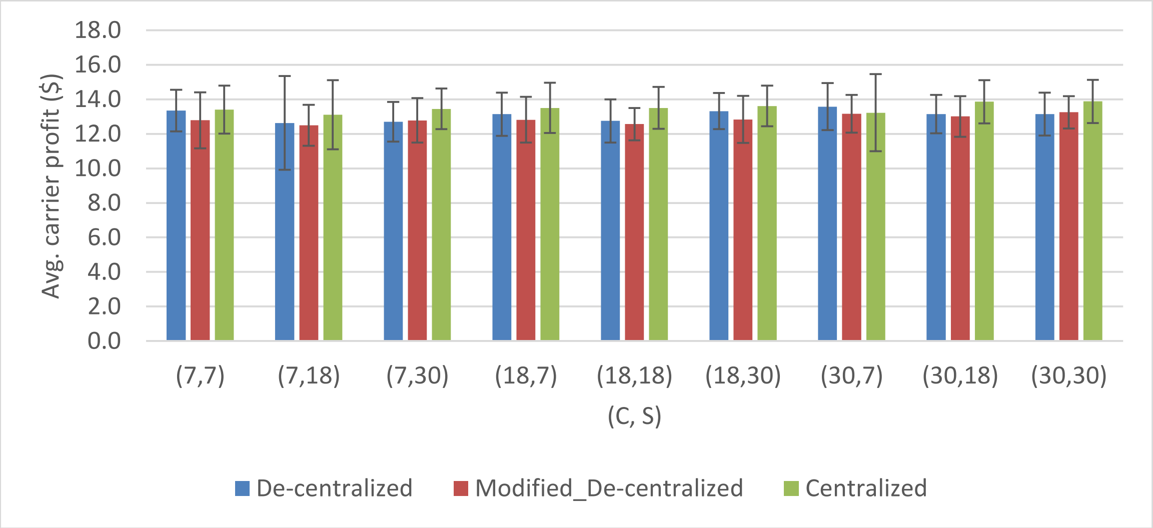

Figure 3: Average carrier profit per tick in all scenarios.

By contrast, the results in Figure 4 show that senders on average save the most if the platform uses the modified-decentralized matchmaking mode because senders tend to match with a carrier closer to them because of the carrier awareness radius leading to higher savings. For a given destination, a carrier will charge a sender less fees compared to a match where the carrier is far away from the sender which could be a situation in a decentralized matchmaking because of the lack of a carrier awareness radius. With centralized matchmaking, the platform matches carriers to senders to maximize its profit, which is a percentage of the fee that the carrier charges to make a delivery. Therefore, the platform will seek to match carriers such that they earn high income. The senders' objective of saving money is not considered by the platform, and as a result, the platform's matchmaking algorithm does not yield the best results for senders.

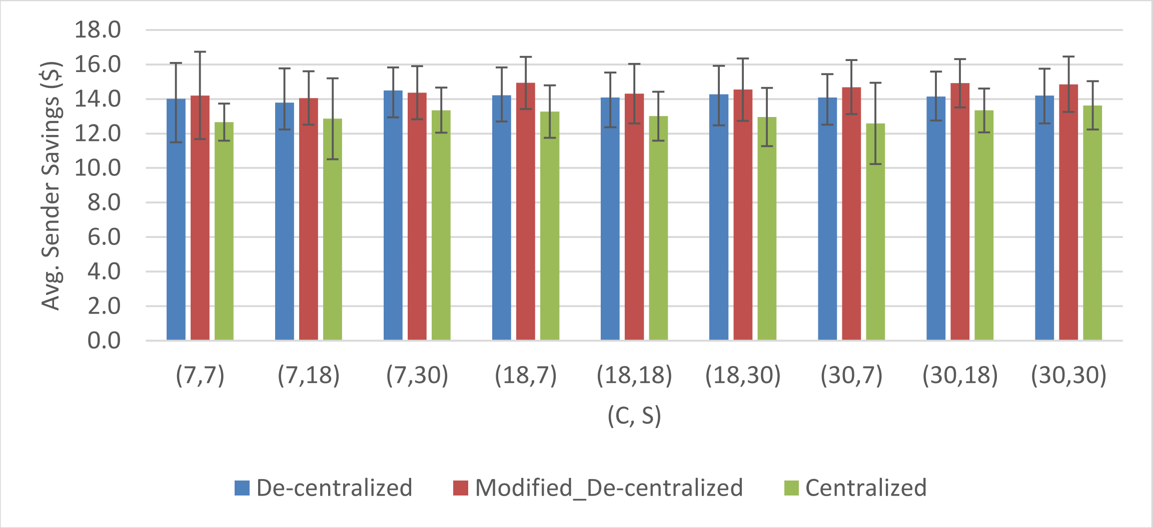

Figure 4: Average sender savings per tick in all scenarios.

4 CONCLUSION

This research used an agent-based model to study the impact of centralized and decentralized mediator control over matchmaking in a crowd-shipping platform. Results from these preliminary experiments indicate that centralized matchmaking performs best overall in terms of maximizing platform revenue, protecting the platform from early failure when there are few initial participants, and increasing the total number of participating senders and carriers over time. The only disadvantage associated with centralized matchmaking that was observed in these experiments was slightly less average daily savings for senders. However, these results are based on the assumptions underlying the specification of the matchmaking algorithm. Specifically, the algorithm was designed to maximize platform revenue, subject to matchmaking that always met the carriers' profit expectations. In other words, the algorithm presented no downsides for the carriers. In reality, empirical research on crowdsourcing systems indicates that platforms do not necessarily consider the participants' individual goals when making assignments, leading to matches that are profitable for the platform but potentially undesirable for the participants. Ongoing development of the model described in this paper will include modifications to the matchmaking algorithm to make it more similar to those of existing centralized platforms, as well as incorporating human behavior data to inform more realistic behavior and decision making of the carrier and sender agents. Specifically, carrier and sender decisions to disintermediate (i.e., "go around" the platform when it assigns them to unfavorable matches) and the resulting impacts on platform performance will be interesting to explore.

ACKNOWLEDGMENTS

This material is based on work supported by The National Science Foundation under Award No. 2046632.

REFERENCES

Calo, R., and A. Rosenblat. 2017. “The Taking Economy: Uber, Information, and Power". 117 Columbia Law Review 1623. https://digitalcommons.law.uw.edu/faculty-articles/47, accessed 15th August.

Carroll, J. M., and V. Bellotti. 2015. “Creating Value Together: The Emerging Design Space of Peer-to-Peer Currency and Exchange". In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, New York, NY, USA: Association for Computing Machinery, 1500-1510.

Castillo, V. E., J. E. Bell, W. J. Rose, and A. M. Rodrigues. 2018. “Crowdsourcing Last Mile Delivery: Strategic Implications and Future Research Directions". *Journal of Business Logistics* 39(1):7–25.

Chen, C., S. Cheng, A. Gunawan, A. Misra, K. Dasgupta, and D. Chander. 2014. “TRACCS: A Framework for Trajectory-Aware Coordinated Urban Crowd-Sourcing". In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing* 2:30–40.

Chen, P., and S. M. Chankov. 2017. “Crowdsourced Delivery for Last-Mile Distribution: An Agent-Based Modelling and Simulation Approach". In *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*. 1271-1275.

Cleophas, C., C. Cottrill, J. F. Ehmke, and K. Tierney. 2019. “Collaborative Urban Transportation: Recent Advances in Theory and Practice". *European Journal of Operational Research* 273(3):801–816.

Devari, A., A. G. Nikolaev, and Q. He. 2017. “Crowdsourcing the Last Mile Delivery of Online Orders by Exploiting the Social Networks of Retail Store Customers". *Transportation Research Part E: Logistics and Transportation Review* 105:105–122.

Dijck, J. V., T. Poell, and M. D. Waal. 2018. "*The Platform Society: Public Values in a Connective World"*. Oxford, England: Oxford University Press.

Dillahunt, T. R., X. Wang, E. Wheeler, H. F. Cheng, B. Hecht, and H. Zhu. 2017. “The Sharing Economy in Computing: A Systematic Literature Review". In *Proceedings of the ACM on Human-Computer Interaction* 1:1-26.

Evans, D. S., and R. Schmalensee. 2010. “Failure to Launch: Critical Mass in Platform Businesses". *Review of Network Economics* 9(4).

Frenken, K. 2017. “Political Economies and Environmental Futures for the Sharing Economy". *Philosophical Transactions of the Royal Society A: Mathematical, Physical, and Engineering Sciences* 375(2095).

Gdowska, K., A. Viana, and J. P. Pedroso. 2018. “Stochastic Last-Mile Delivery with Crowdshipping". *Transportation Research Procedia* 30:90–100.

Gerber, C. 2021. “Community Building on Crowdwork Platforms: Autonomy and Control of Online Workers?". *Competition & Change* 25(2):190–211.

Ghezzi, A., D. Gabelloni, A. Martini, and A. Natalicchio. 2018. “Crowdsourcing: A Review and Suggestions for Future Research". *International Journal of Management Reviews* 20(2):343–63.

Grimm, V., U. Berger, D. L. DeAngelis, J. G. Polhill, J. Giske, and S. F. Railsback. 2010. “The ODD Protocol: A Review and First Update". *Ecological Modelling* 221(23):2760–2768.

Gruenbaum, R. 2015. “Critical Mass". In *Making Social Technologies Work: Leveraging the Power and Managing Perils of Social Technologies in Business*, *Palgrave Pocket Consultants*, edited by R. Gruenbaum, 142-146. London: Palgrave Macmillan UK.

Howe, J. 2006. “The Rise of Crowdsourcing". *Wired Magazine* (14):176-183.

Kohler, T. 2015. “Crowdsourcing-Based Business Models: How to Create and Capture Value". *California Management Review* 57(4):63–84.

Lampinen, A., K. Huotari, and C. Cheshire. 2015. “Challenges to Participation in the Sharing Economy: The Case of Local Online Peer-to-Peer Exchange in a Single Parents’ Network". *Interaction Design and Architecture(s)* (24):16–32.

Le, T. V., A. Stathopoulos, T. V. Woensel, and S. V. Ukkusuri. 2019. “Supply, Demand, Operations, and Management of Crowd-Shipping Services: A Review and Empirical Evidence". *Transportation Research Part C: Emerging Technologies* 103:83–103.

Le, T. V., S. V. Ukkusuri, J. Xue, and T. V. Woensel. 2021. “Designing Pricing and Compensation Schemes by Integrating Matching and Routing Models for Crowd-Shipping Systems". *Transportation Research Part E: Logistics and Transportation Review* 149:102209.

Lee, M. K., D. Kusbit, E. Metsky, and L. Dabbish. 2015. “Working with Machines: The Impact of Algorithmic and Data-Driven Management on Human Workers". In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, *CHI ’15*, 1603– 1612. New York, NY, USA: Association for Computing Machinery.

Light, A., and C. Miskelly. 2015. “Sharing Economy vs Sharing Cultures? Designing for Social, Economic and Environmental Good". *Interaction Design and Architecture(s)* (24):49–62.

Mittal, A., N. O. Gibson, C. C. Krejci, and A. A. Marusak. 2021. “Crowd-Shipping for Urban Food Rescue Logistics". *International Journal of Physical Distribution & Logistics Management* 51(5):486–507.

Möhlmann, M., and L. Zalmanson. 2017. "Hands on the Wheel: Navigating Algorithmic Management and Uber Drivers' Autonomy ". In *Proceedings of the International Conference on Informational Systems (ICIS)*, December 10th-13th, Seoul, South Korea.

Möhlmann, M., L. Zalmanson, O. Henfridsson, and R. W. Gregory. 2021. “Algorithmic Management of Work on Online Labor Platforms: When Matching Meets Control". *MIS Quarterly* 45(4):1999–2022.

Nieto-Isaza, S., P. Fontaine, and S. Minner. 2022. “The Value of Stochastic Crowd Resources and Strategic Location of Mini-Depots for Last-Mile Delivery: A Benders Decomposition Approach". *Transportation Research Part B: Methodological* 157:62–79.

Ntouros, V., H. Kouki, and V. Vlachokyriakos. 2021. “Designing Sharing Economy Platforms through a ‘Solidarity HCI’ Lens". *Proceedings of the ACM on Human-Computer Interaction* 5:1-25.

Punel, A., and A. Stathopoulos. 2017. “Modeling the Acceptability of Crowdsourced Goods Deliveries: Role of Context and Experience Effects". *Transportation Research Part E: Logistics and Transportation Review* 105:18–38.

Rougès, J. F., and B. Montreuil. 2014. “Crowdsourcing Delivery: New Interconnected Business Models to Reinvent Delivery". *1st International Physical Internet Conference*, Québec City, Canada, 1-19.

Scholz, T. 2016. "Platform Cooperativism. Challenging the Corporate Sharing Economy". New York, NY: Rosa Luxemburg Stiftung.

Schor, J. B., and W. Attwood-Charles. 2017. “The ‘Sharing’ Economy: Labor, Inequality, and Social Connection on for-Profit Platforms". *Sociology Compass* 11(8):e12493.

Sutherland, W., and M. H. Jarrahi. 2018. “The Sharing Economy and Digital Platforms: A Review and Research Agenda". *International Journal of Information Management* 43:328–341.

Wazny, K. 2017. “‘Crowdsourcing’ Ten Years in: A Review". *Journal of Global Health* 7(2):020602.

Westelaken, M., and Y. Zhang. 2017. "An Agent-Based Model for Feasibility and Diffusion of Crowd Shipping". In *29th Benelux Conference of Artificial Intelligence*, November 8th - 9th, Groningen, Netherlands, 419.

Writz, J., K. K. F. So, M. A. Mody, S. Q. Liu, and H. E. H. Chun. 2019. “Platforms in the Peer-to-Peer Sharing Economy" *Journal of Service Management* 30(4):452–483.

AUTHOR BIOGRAPHIES

PREETAM KULKARNI is a PhD student in the Department of Industrial, Manufacturing, & Systems Engineering at the University of Texas at Arlington. He obtained his B.E. in Mechanical Engineering from University of Visvesvaraya College of Engineering, India and a M.S. in Industrial Engineering from Iowa State University. His research interests include agent-based modelling, optimization and applications of data analytics in operations management. His email address is preetam.kulkarni@uta.edu.

CAROLINE C. KREJCI is an Assistant Professor of Industrial, Manufacturing, & Systems Engineering at The University of Texas at Arlington. She holds a Ph.D. in Industrial Engineering from the University of Washington. Her research is focused on the development of quantitative methodologies for the analysis and sustainable management of sociotechnical systems, including supply networks and production systems. Her email address is caroline.krejci@uta.edu.