Understanding Gig Worker Resistance and Cooperation under Algorithmic Management

SUBMISSION 1823

As algorithmic management continues to shape coordination and labor dynamics on both on-demand platforms and in traditional workplaces, understanding how workers and managers collectively interact with these systems is critical for navigating the evolving labor landscape. While existing research has documented instances of worker resistance–such as task selectivity on ride-hailing platforms–this paper analyzes collaborative and resistive behaviors, quantifying their welfare implications for both platforms and workers through a game-theoretic framework and a behavioral experiment. By integrating human manager biases into the model, we uncover conditions where these biases positively influence group outcomes and overall welfare. Our analysis extends to complex, real-world scenarios involving fluctuating demand and interactions between multiple workers, offering a more accurate reflection of gig platform environments. Additionally, we conduct a behavioral experiment with full-time managers to explore the interplay of these biases and investigate workers' perceptions of human versus algorithmic management. This research sheds light on how collaborative decision-making processes unfold between human and algorithmic managers, providing practical insights for platforms to consider towards improving coordination between itself and the workers it employs.

Contents

Manuscript submitted for review to the 26th ACM Conference on Economics & Computation (EC'25).

1 Introduction

Algorithmic management is transforming how workplaces operate, from logistics and transportation to retail and customer service. By automating tasks like scheduling, task assignment, and performance evaluation, platforms such as Amazon and Uber optimize operations with unprecedented speed and scale [Delfanti, 2021, Rosenblat and Stark, 2016]. These systems are designed to process vast amounts of real-time data, making them particularly effective in managing large-scale, dynamic environments. However, the shift toward algorithmic management introduces complex challenges, especially in understanding how workers interact with these systems and adapt to algorithmic decision-making.

Unlike traditional management, where human managers rely on communication and subjective judgment, algorithmic systems impose a different mode of authority. Workers often perceive algorithms as impersonal, opaque, or rigid, which can lead to disengagement, resistance, or strategic gaming of the system. For instance, workers on gig platforms have adopted creative strategies, such as manipulating location-based algorithms to gain favorable assignments [Borak, 2022, Brooks, 2020]. At the same time, algorithm aversion–a phenomenon where workers prefer human managers over AI, even when the AI is demonstrably more accurate–further complicates the adoption of these systems [Dietvorst et al., 2015, Lee, 2018]. Uber drivers were reported to be frustrated with algorithmic management and their resentment can lead them to behave subversively with the potential to cause real harm to the company [Möhlmann and Henfridsson, 2019].

Adding to this complexity is the role of human biases in management. Human managers often rely on cognitive shortcuts, such as recency and anchoring biases, which can distort decision-making [Bazerman and Moore, 2012, Tversky and Kahneman, 1974]. While these biases are often seen as inefficiencies, they may occasionally benefit task allocation by reinforcing positive behaviors or adapting to social nuances. Conversely, algorithms are designed to minimize such biases but are not immune to their own forms of inefficiency, especially when trained on biased data. The interplay between human biases, algorithmic decision-making, and worker behavior remains poorly understood, raising important questions about when and how algorithms should augment or replace human managers.

We address these questions by examining how workers respond to algorithmic versus human management and exploring the implications of managerial biases in task allocation. Our game-theoretic model characterizes task assignment dynamics, focusing on how workers strategically respond to managerial decisions while incorporating key biases, *recency* and *anchoring*. Additionally, we conduct a behavioral experiment with professional managers to measure these biases and their effects on decision-making and platform outcomes.

Our findings reveal a nuanced picture. While algorithmic management excels in optimizing efficiency, it often struggles to navigate the social and strategic dimensions of human interactions. Human biases, though imperfect, can sometimes enhance collaboration by providing immediate feedback and reinforcing desirable behaviors. However, these benefits are highly context-dependent and can lead to inefficiencies in settings where fairness and transparency are critical. By modeling and empirically analyzing these dynamics, our study sheds light on the conditions under which human and algorithmic management are most effective, offering insights for platform design.

1.1 Related Work and Contributions

Our work contributes to several research streams by exploring how workers interact with algorithmic and human managers, uncovering biases in managerial decision-making, and offering insights into platform design to improve cooperation between platforms and workers.

Human managers often exhibit cognitive biases, such as anchoring and availability heuristics, which can distort task assignments and lead to inconsistent or unfair outcomes [Carter et al., 2007, Tversky and Kahneman, 1974]. For example, prior performance ratings can disproportionately influence promotions or task allocations, skewing managerial decisions. This is particularly concerning in gig work, where platform algorithms have been criticized for reinforcing disparities and disproportionately impacting vulnerable workers [Ma et al., 2022, Munoz et al., 2022]. Although much research has focused on identifying these biases, our behavioral experiment quantifies their prevalence in task assignments and compares their effects under human and algorithmic management.

Beyond biases, workers' perceptions of algorithmic systems critically shape their engagement with and resistance to these platforms. The literature on algorithm aversion has documented how concerns over interpretability and trust limit the adoption of AI-based recommendations [Dietvorst et al., 2015, Mok et al., 2023]. Workers often perceive AI as efficient and objective but lacking the empathy and adaptability associated with human managers [Lee, 2018, Park et al., 2021]. In contrast, human managers are seen as more flexible and capable of accounting for personal circumstances, even though their decisions are accompanied by biases and inefficiencies [Ashktorab et al., 2020, Zhang et al., 2021]. These contrasting perceptions create challenges for hybrid management systems, where balancing efficiency with worker trust becomes critical. Understanding how these perceptions affect worker behavior is essential for designing platforms that better leverage AI's capabilities.

At the same time, the control exerted by algorithmic management often conflicts with the autonomy highly valued by gig workers [Jarrahi et al., 2020, Kusk and Nouwens, 2022]. Workers frequently engage in strategic resistance to reclaim autonomy, employing tactics such as manipulating proximity-based assignments and cherry-picking tasks [Borak, 2022, Brooks, 2020]. These actions reflect broader tensions between workers and platforms, where algorithms simultaneously serve as tools of control and targets of resistance [Cameron and Rahman, 2022, Hao and Freischlad, 2022]. Our game-theoretic model extends this work by examining the mechanisms driving resistance and proposing platform-level interventions to address its root causes.

In addition to resistance, workers often adapt their behaviors in response to algorithmic systems, engaging in anticipatory compliance and strategic gaming to optimize outcomes [Bucher et al., 2021, Ramesh et al., 2023]. For example, gig workers may undervalue their labor to avoid punitive ratings or underperform strategically to minimize unfavorable shifts [Rosenblat and Stark, 2016]. These adaptations illustrate the ongoing negotiation between algorithmic control and worker autonomy, highlighting the need for platforms to design systems that encourage productive behaviors without imposing undue restrictions. Previous research has used game theory to model how incentive structures, such as bonus strategies and matching algorithms, influence worker decisions and platform efficiency [Liu and et al., 2019, Mai et al., 2023]. For instance, contingent bonus strategies that reward consistent participation can create competition and inefficiencies, whereas fixed bonuses may improve worker satisfaction but reduce utilization rates [Liu and et al., 2019]. Our work builds on these insights by examining how platform design can mitigate worker resistance while fostering cooperation.

The rise of algorithmic management systems has also prompted a growing body of research comparing their effects to those of human managers. While algorithmic systems are often praised for their efficiency and objectivity [Lee et al., 2015], workers perceive them as lacking the empathy and flexibility that human managers provide [Bucher et al., 2021, Fumagalli et al., 2022]. For instance, studies have shown that while workers view algorithmic evaluations as more objective, they often feel alienated due to the lack of transparency and human intuition [Cameron and Rahman, 2022, Lee, 2018]. This trade-off between efficiency and emotional engagement underscores the need to design hybrid systems that leverage the strengths of both human and algorithmic management.

Our study bridges these research streams by addressing three key questions. First, how do biases in human and algorithmic management influence task allocation and worker outcomes? Second, what mechanisms underlie worker resistance to algorithmic control, and how can platforms address these challenges? Third, how can platform-level interventions improve cooperation and efficiency in hybrid management systems? By combining behavioral experiments, game-theoretic modeling, and insights into platform design, we offer actionable recommendations for balancing algorithmic efficiency with worker autonomy and engagement.

2 The Game Theoretic Model of Gig Worker-Platform Interactions

In order to represent the dynamic decision-making of both workers and managers on a gig platform, we develop the following model. This model specifically seeks to understand the mechanisms behind "cherry picking" behavior seen on gig platforms. Using backwards induction and simulations, we can identify circumstances in which worker would reject assigned tasks and how preferences, worker performance, and manager biases feed into these calculations.

2.1 Model Setup

Our main model considers a sequential interaction between workers and managers over N periods. For now, we are solely concerned with the case of one manager managing one worker (or this can be considered in a case of many non-interacting workers). The sequence of actions in a given period are as follows: the manager assigns the worker a type of job, the worker chooses whether to accept the assigned job, and the manager is then informed of the worker's decision as well as the performance of the job if accepted. For simplicity, the space of jobs only consist of two types of jobs, type A and type B. Similarly, worker performance on jobs is classified into the two categories of high and low performance.

First, looking at the worker side of this model, for a worker i, we can quantify their preferences at a given time by the parameter, $p_i \in [0, 1]$. This value simply represents the probability that worker i will prefer a job of type A on a given day. If they don't prefer type A, they will prefer type B instead. These preferences are expressed as different utilities that the worker receives by accepting and completing the task. u_2 is the utility received for completing a task the worker prefers, while u_1 the corresponding utility for completing a non-preferred task. **Assumption:** $u_2 > u_1 > 0$. Looking into the future, workers also face a discount factor of $\delta \in (0, 1]$ and a present bias of $\beta \in (0, 1]$.

This randomization of worker preferences is meant to help simulate the myriad of different conditions under which a gig worker might find themselves in, where they might have differing preferences for what types of jobs they want to take on. An example would be a ride-hailing driver preferring longer trips on arbitrary days when they have a longer uninterrupted block of worker, and shorter trips otherwise. This also helps to address why workers cannot directly communicate their preferences to the managers, human or algorithm, as their own preference parameters might not be entirely known to them.

Worker performance operates similarly to worker preferences, where every worker i has two parameters q_{Ai} and q_{Bi}. The parameter q_{Ti} is the probability that a worker will have high performance completing a job of type T (otherwise they will complete the task will low performance). Worker utilities do not depend on their performance in this model, so as a result, will try to optimize actions towards their own preferences (p_i values).

Shifting towards the manager side of the model, managers receive utility to workers completing jobs, which is performance dependent. u_H is received whenever a worker completes a job with high performance, u_L is likewise received for a worker completing a job with low performance. **Assumption:** $u_H > u_L > -u_H$ and $u_H > 0$ (*notice that u_L can be negative, although not too negative so that low performance penalized more than high performance rewards*). Managers don't know the parameter values of p_i, q_{Ai}, and q_{Bi} for their workers, so their choices of job assignment has a twofold purpose: to learn preferences and performances of their workers, and to use that knowledge to optimize acceptances and high performance rates.

Similarly, for workers, in order to optimize their own utility, they can use their knowledge of how managers might act in order to get managers to suggest their preferred types of jobs. What emerges from this framework is a situation where workers attempt to imperfectly signal their preferences to platforms, against issues of performance and short-term utility gains.

2.2 Theoretical Results

In order to detail some basic results from this model, we will first state the assumptions included with this main model:

(1) **Worker preferences and performance stays fixed over the N periods** (i.e. p_i, q_{Ai}, and q_{Bi} have no time dependence). This is a fairly reasonable assumption to make over a short time period implying there is no potential for learning or preference changes.
(2) **There is an infinite supply of both job types available.** Functionally, this assumption means that managers always can assign a job of either type, so we can focus strictly on decision maximizing behavior of manager.
(3) **Manager behavior assumes that workers will accept a job if and only if that assigned job matched their current preference (which was randomly selected according to p_i).** Manager decisions will be largely dictates by their beliefs on worker parameters, and so can rely on a fairly simplified model to simulate choices with managers as short-sighted decision makers.

With these assumptions, we can directly work out a number of theoretical results for the simple case of a single worker over $N = 2$ periods. *Note that there is no real difference in our choice of analyzing the case a single worker due to assumption (2) of infinite supply.*

First, we solve for the behavior of managers given historical observations of the work up to that point. Managers should choose to assign the job type that would maximize their own expected utility for that period. This calculation of expected utility is done using their beliefs regarding the parameters of the worker (p_i, q_{Ai}, and q_{Bi}) according to their previous observations. To notate previous worker actions, define:

$$X_{it} = \begin{cases} 1 & \text{if worker } i \text{ accepted job A or rejected job B in period } t \\ 0 & \text{otherwise (accepted job B or rejected job A)} \end{cases}$$

and similarly for performance,

$$Y_{it} = \begin{cases} 1 & \text{if worker } i \text{ has high performance in period } t \\ 0 & \text{otherwise (low performance or rejected the job)} \end{cases}$$

$$Z_{it} = \begin{cases} 1 & \text{if worker } i \text{ accepts the assigned job} \\ 0 & \text{otherwise (rejects the assigned job)} \end{cases}$$

Note that by our assumptions regarding preferences, we can record the same value of X_{it} for accepting type A/rejecting type B or vice-versa as in both cases, the manager would conclude that the decision was due to a preference towards type A (or type B in the other case).

By our setup, managers can estimate worker parameters for both preferences and performance using a Beta distribution. In this case, successes (k) are when $X_{it} = 1$ and failures ($n - k$) are when $X_{it} = 0$, aka whenever the worker prefers a job of type A. (For performance, success is equated to high performance of the job type and failure is low performance of the job type). Thus, by taking the expected value of the distribution Beta$(k+1, n-k+1)$, we can get the manager's estimate for the parameters. The expected value for a Beta distribution is $\frac{\alpha}{\alpha+\beta} = \frac{k+1}{k+1+n-k+1} = \frac{k+1}{n+2}$.

So, assuming *perfect memory* managers, we have the following estimates after period n:

$$\hat{p_i} = \frac{1 + \sum_{t=1}^{n} X_{it}}{2+n} \tag{1}$$

$$\hat{q}_{Ai} = \frac{1 + \sum_{t=1}^{n} X_{it}Y_{it}Z_{it}}{2 + \sum_{t=1}^{n} X_{it}Z_{it}}, \ \hat{q}_{Bi} = \frac{1 + \sum_{t=1}^{n}(1-X_{it})Y_{it}Z_{it}}{2 + \sum_{t=1}^{n}(1-X_{it})Z_{it}} \tag{2}$$

where the $X_{it}Z_{it}$ and $(1-X_{it})Z_{it}$ terms in the summation function as indicator variables for when worker i accepted job A or job B respectively.

With these parameter estimates, we can mathematically solve for the manager's behavior that maximized the platform's expected utility as done in the appendix.

This gives the following theorem for manager behavior:

Theorem 2.1. *A manager optimizes their expected utility by assign a job of type A in period n to worker i, given their estimates $\hat{p}_i$, $\hat{q_{Ai}}$, $\hat{q_{Bi}}$, if the following condition holds.*

$$\hat{p}_i\hat{q}_{Ai} + (p-1)\hat{q}_{Bi} \geq (1-2\hat{p}_i)\frac{u_L}{u_H - u_L} \tag{3}$$

Otherwise, the manager should assign a job of type B.

Now that we have a proper model of the managers' behavior, we can deduce what a worker's actions should be given this model of the manager using backwards induction. For sake of demonstrating the key features of this model, we will first solve in the case of **one** worker over **two** periods. Additionally, we will set $u_H = 1$ to simplify expressions. So by our earlier assumptions, this means that $-1 < u_L < 1$.

First, we can note that in the second period (the last time period), the worker should always accept whatever job given as there is no additional utility to be gained from rejection. Thus, we are only concerned with the decision of the worker in the first period. By Theorem 3.1, the manager should assign job type A in the first period (all estimates are currently 0.5). Then, by using backwards induction, we look at the subgame states from each possible outcome following the initial assignment and can evaluate the expected utility from those states.

If the worker chooses to accept job A in the first period and subsequently has high performance, then the manager updates their beliefs to $\hat{p} = 2/3$, $\hat{q_A} = 2/3$, and $\hat{q_B} = 1/3$ (unchanged). Plugging these values into theorem 3.1, we get the condition that the manager will assign a job of type A if $u_L \geq -2.6$, and by our assumptions on u_L, this means the manager will always assign a job of type A in this case.

Similarly, we can work out the estimates in the case where the worker accepts the job in period 1, but has low performance. The estimates are detailed in Figure 1 and after plugging into the condition, we find that the manager assigns job A again if $u_L \geq -0.2$, otherwise job B will be assigned.

Finally, repeating this process in the case if the worker rejects the period 1 job assignment, we get that the manager assigns job A if $u_L \leq -1$. Thus, due to our assumptions, the manager should always assign job B in this scenario.

Now that we have determined the results of the subgames, we can use this to calculate the expected utility of the worker either accepting or rejecting the initial job A assigned, choosing the higher utility outcome.

We first consider the case where $u_L \geq -0.2$, so the main difference that we see is a difference assignment in the accept, but low performance subgame. Let the utility gained from accepting the assigned job (in this case A) be denoted by u ($u = u_H$ if the worker prefers A in the period 1,

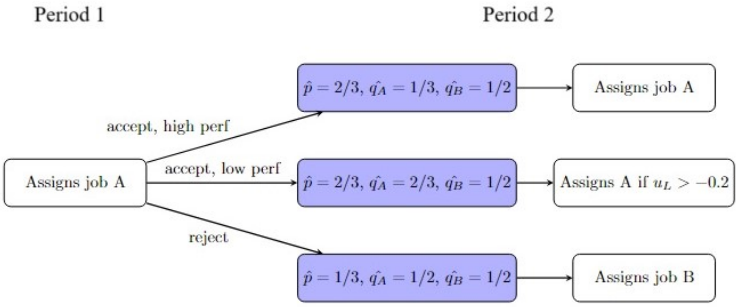

Period 1
Period 2
Assigns job A
accept, high perf
accept, low perf
reject
$\hat{p} = 2/3$, $\hat{q_A} = 1/3$, $\hat{q_B} = 1/2$
$\hat{p} = 2/3$, $\hat{q_A} = 2/3$, $\hat{q_B} = 1/2$
$\hat{p} = 1/3$, $\hat{q_A} = 1/2$, $\hat{q_B} = 1/2$
Assigns job A
Assigns A if $u_L > -0.2$
Assigns job B

Fig. 1. Diagram of backwards induction for the two period model.

$u = u_L$ if they prefer B instead). Then, the worker should accept the assigned job A if the following equality holds:

$$\mathbb{E}[U|\text{accept A, period 1}] \geq \mathbb{E}[U|\text{reject A, period 1}]$$
$$u + \beta\delta[pu_2 + (1-p)u_1] \geq \beta\delta[pu_1 + (1-p)u_2]$$
$$\dots$$
$$u \geq \beta\delta(u_2 - u_1)(1 - 2p) \tag{4}$$

Now repeating for the case where $u_L < -0.2$, our inequality instead becomes:

$$u + \beta\delta[q_A(pu_2 + (1-p)u_1) + (1-q_A)((1-p)u_2 + pu_1)] \geq \beta\delta[pu_1 + (1-p)u_2]$$
$$\dots$$
$$u \geq \beta\delta q_A(u_2 - u_1)(1 - 2p) \tag{5}$$

Thus, plugging in the appropriate values of u under the relevant conditions into equations (4) and (5), we arrive at the theorem:

Theorem 2.2. *In the two period game under a memory-perfect manager with a single worker, the worker will choose to accept the job A assigned in period 1 under the following conditions:*

- *If $u_L \geq -0.2$*
  - *If $X_1 = 1$ (prefers A in period 1), $Z_1 = 1$ aka accept job assignment A.*
  - *If $X_1 = 0$, $u_1 \geq \beta\delta(u_2 - u_1)(1 - 2p)$*
- *If $u_L < -0.2$*
  - *If $X_1 = 1$, $Z_1 = 1$ aka accept job assignment A.*
  - *If $X_1 = 0$, $u_1 \geq \beta\delta q_A(u_2 - u_1)(1 - 2p)$*

Regardless of the result, $Z_2 = 1$ aka the worker so always accept the assignment in the second and last period.

Note that specifically in the $X_1 = 1$ cases, the inequality from equation (4) becomes

$$u_2 \geq \beta\delta(u_2 - u_1)(1 - 2p)$$

which we can note always holds in with our assumptions since,

$$\beta\delta(u_2 - u_1)(1 - 2p) \leq u_2 - u_1 < u_2$$

since $\beta, \delta, 1 - 2p \leq 1$. Thus, the worker should always accept in the case that their assigned job matches their current preference.

Looking at the cases where $X_1 = 0$, so the assigned job A doesn't match the worker's current preference, we can see that if $p \geq 1/2$, then the worker should accept the job assignment even though it isn't their preferred type. However in the case where $p < 1/2$, then it can become better for the worker to reject the assignment if the difference between u_2 and u_1 is large enough. The main difference between the different u_L conditions in this case, is that when u_L is negative enough, their is an additional q_A term in front of the right hand expression. As a result, there must be an even higher difference between u_2 and u_1 to get the worker to reject the assigned job, since now the manager will still assign job B if the performance is low on task A.

We can observe an inherent tension in this model between the short-term gains of doing jobs and the long-term strategy of ensuring that the manager has an accurate estimate of your p value. In the case where both align (i.e. $p \geq 1/2$, so accepting job type A signals the correct direction), the worker is always incentives to accept the task. However in cases where these goals are opposed, such as when the worker doesn't prefer job type A and has $p < 1/2$, we can see that a high enough difference between u_2 and u_1 is enough for a worker to forego short-term gains of accepting an undesired task, in order to correctly signal to the manager in the hopes of getting more accurate assignments in the future. So, we can see how this heavy simplified, two-period model predicts the worker behavior of cherry-picking job assignment seen on real-life gig platforms.

Note that due to the non-interactivity of workers and the assumption of unlimited supply of jobs, nothing in the model changes by adding more workers. So, without loss of generality, we can extend all of the results we obtained here to an arbitrary n worker model.

3 Experimental Design

In order to further utilize our theoretical model in analyzing different potential outcomes on gig platforms under human versus algorithmic management, we need to develop an understanding of the behavior of human managers in this setting. To collect this data, we devise an online behavioral experiment in which participants play a role of a manager who allocates tasks to virtual on-demand workers on a gig platform. The pre-registration of our experiment is accessible through https://aspredicted.org/KT7_S8K.

The game is played over 9 rounds, during which participants are tasked with assigning jobs of either type A or type B to a wide variety of different workers, similar to the setup in the model. Each round sees the participant matched with predetermined worker(s) that have a fixed underlying preference and performance rate. Over the course of 10 simulated days in the round, the participant must learn the preferences and performance parameters of workers (p_i, q_{Ai}, q_{Bi}). Each day, the participant can assign either job type A or job type B to each worker, who then decided whether or not to accept their assigned task. The participant is then asked to provide estimates for these parameters after the 5th day and after the 10th day.

For the first three rounds, participants are only tasked with assigning tasks to one worker in a setting without results of performance. So the worker during these rounds is only concerned with if the singular worker accepts or rejects each job, and thus formulates their estimation entirely from these observations. This is useful to both gradually introduce the mechanics of the game to the participants and also have experimental data on estimation before introducing extra complications.

Then for the final six rounds, participants must now manage two workers and have to deal with the additional complication of performance. Now, in addition to estimating parameters, managers are tasked with also maximizing their score over that round. If a worker chooses to accept their assigned job, then they will either complete it with high or low performance, which give different amounts of score. If the worker rejects the task, the manager gets zero additional score. Throughout these rounds, participants are forced to juggle learning workers' preferences and performance parameters and maximizing their own score, similar to an exploit vs explore framework. Furthermore, the addition of extra parameters, in the form of performance metrics, and an extra worker serve to further hinder their ability to successfully estimate parameters and strengthen the effects of any potential biases.

In addition, we also vary parameters and worker strategies significantly from round to round to collect data and behavioral information over a wide range of circumstances. Specifically, for most workers, we pre-draw outcomes according to their designated preferences such that for some rounds, worker behavior is similar between the first five days and the last five days in terms of preferences. However in other rounds, we change the proportion of type A jobs preferred to type B jobs between the first and last five days of the round, although in such a way that is still valid given their underlying parameters.

Besides changing values of p, q_A, and q_B, workers are also classified into one of two categories: *perfectly-signaling* and *strategic*. Perfect-signaling workers behave exactly according to their underlying preferences, so if they prefer a job of type A on any day, they will accept jobs of type A and reject jobs of type B, and vice versa for preferring type B. On the other hand, strategic workers behave like perfectly-signaling workers for the first six days, before then accepting any job that is assigned to them regardless of their preference. The idea of this worker is to signal their preferences for the beginning of the round, and then switch to maximizing their own utility by accepting every assignment. They are under the assumption that rejecting any future jobs to continue signaling doesn't have a large enough benefit over the remaining days to offset the short-term gain of accepting a job they do not like.

Figure 2 displays the underlying parameters of each worker over every period and what type they are. The order listed is arbitrary as order was randomized for participants during the actual survey. Figure 3 illustrates an overview of the general design and flow of the experiment.

Round	p	Type
1	0.2	PS
2	0.5	PS
3	0.2	S

Round	p_1	q_{A1}	q_{B1}	Type 1	p_2	q_{A2}	q_{B2}	Type 2
4	0.7	0.8	0.2	PS	0.7	0.2	0.8	PS
5	0.8	0.4	0.6	PS	0.2	0.4	0.6	PS
6	0.5	0.5	0.5	PS	0.2	0.2	0.8	PS
7	0.3	0.2	0.8	S	0.3	0.6	0.6	PS
8	0.9	0.4	0.6	PS	0.4	0.4	0.6	S
9	0.3	0.5	0.5	S	0.8	0.9	0.1	S

Fig. 2. Table of the underlying parameters for each worker in the nine game rounds. Note: PS is short for perfect-signaling and S is short for strategic.

For each of these rounds, we record all of the data collected during the round, such as the task assigned to each worker, whether it was accepted, performance on the task, and overall score tracking, which will be helpful for variable construction during the analysis. Additionally, we collect survey data via the two sections at the end, illustrated in figure 3, which record participants chosen strategies as well as some qualitative questions regarding their perceptions of algorithmic vs. human management from a worker's perspective.

With this rich amount of data we have access to, the main question we are concerned with are:

- What biases can we see are present in human management decision-making and,
- To see to what extent these biases are present in human managers.

Our study has a 2 × 2 factorial design. One condition controls if a history of the past acceptances and performances of the worker are shown, which can be either *History (past 5 days are shown)* or *No History.* The second condition concerns scoring for high vs. low performance on accepted jobs.

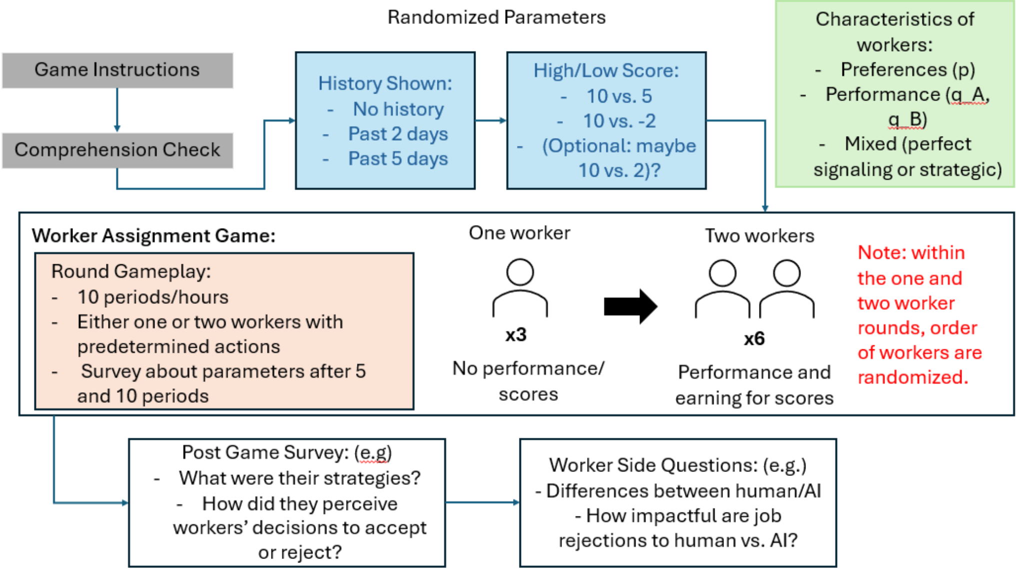

Randomized Parameters
Game Instructions
Comprehension Check
History Shown:
- No history
- Past 2 days
- Past 5 days
High/Low Score:
- 10 vs. 5
- 10 vs. -2
- (Optional: maybe 10 vs. 2)?
Characteristics of workers:
- Preferences (p)
- Performance (q_A, q_B)
- Mixed (perfect signaling or strategic)
Worker Assignment Game:
Round Gameplay:
- 10 periods/hours
- Either one or two workers with predetermined actions
- Survey about parameters after 5 and 10 periods
One worker
x3
No performance/ scores
Two workers
x6
Performance and earning for scores
Note: within the one and two worker rounds, order of workers are randomized.
Post Game Survey: (e.g)
- What were their strategies?
- How did they perceive workers' decisions to accept or reject?
Worker Side Questions: (e.g.)
- Differences between human/AI
- How impactful are job rejections to human vs. AI?

Fig. 3. Illustration of experimental design and flow

In both conditions, high performance will always reward 10 points. The reward for low performance is conditioned on the following two possibilities: *Small Reward (gives 5 points)* or *Small Penalty (gives* -2 *points)*.

With these conditions, we see to test the following hypotheses:

- **H1:** Participants that do not have a history log available will exhibit more dramatic biases in their preference estimates.
- **H2:** Participants with low performance penalties will exhibit more dramatic biases in their preference estimates.
- **H3:** Participants that do not have a history log available will see lower performance scores on average.
- **H4:** Participants with low performance penalties will place greater importance on exploring and discovering worker preferences before exploiting their current knowledge (this is tested via a post-game survey response question).

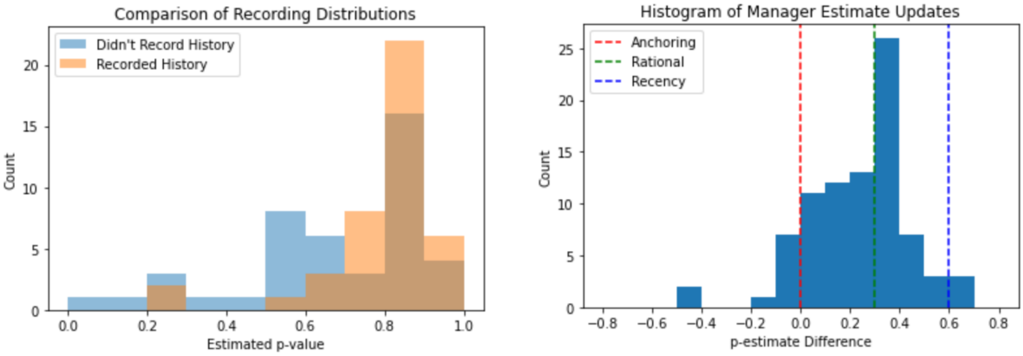
4 Experimental Results

We recruited a total of 112 full-time managers, all of whom were enrolled in a part-time graduate program in business administration at a U.S. university, to participate in this study. At the conclusion of the experimental period, 86 participants successfully completed the study. Given that these participants hold managerial positions in their respective organizations, their actions and decision-making processes during the experiment are expected to be reflective of the biases inherent in real-world managerial behavior.

Figure 4 shows the breakdown of the four experimental conditions. Despite the imbalanced composition, particularly noticeable in the “history" condition, sufficient data remains available for robust analysis when focusing on the other conditions. This suggests that while the distribution issue should be noted, it does not critically undermine the validity of the experimental findings within the scope of the remaining conditions.

	History	No History	Total
Small Reward	28	14	42
Small Penalty	19	25	44
Total	47	39	86

Fig. 4. Table of counts for the four different treatment conditions in the experiment



(a) Distributions of p-estimates for employee 1 during the fifth round.

(b) Distribution of the p-estimate updates for employee 1 of round 6.

Fig. 5. Experimental distributions for p-estimates.

An unanticipated finding in our data was that a significantly higher percentage of participants than expected reported using a strategy of recording all observed data, which theoretically should have resulted in perfect accuracy. Upon reviewing the estimation strategies provided by participants, we identified that approximately 48% of the responses claimed to employ this strategy.

Interestingly, despite the use of such strategies, the impact on the overall data distribution appears minimal. As shown in Figure 5a, rather than converging on a single value, the estimates exhibit variability, similar to the responses from participants who did not report recording their observations. This suggests that either participant error or additional factors influencing their estimation process prevented the perfect accuracy one might expect from this strategy. Notably, we observe a higher variance in the responses from those who did not record their observations, indicating that while the reported strategy may have improved consistency for some participants, it was not uniformly applied or effective.

4.1 Bias Estimation

In addressing the issue of quantifying managerial biases, we define biases in our context as behaviors that deviate from optimal decisions given prior information. We focus on two key cognitive biases that are prevalent in decision-making processes:

- **Recency Bias** refers to the tendency for individuals to prioritize and give disproportionate weight to more recently observed data points when making decisions, often leading to an overemphasis on short-term trends.
- **Anchoring Bias** occurs when individuals rely heavily on an initial reference point (or “anchor") and inadequately adjust their beliefs in response to new information, thus remaining closer to their prior beliefs even when evidence suggests otherwise.

Below is an instructive figure for Round 6, where employee 1 has a significant preference difference between the first 5 days compared to the last 5 days. Note that, under complete anchoring bias for the first 5 days, where new observations have no effect on the estimate, the update will always be zero. On the other hand, complete recency bias over the 5 day time period will result in an update of roughly double the update value of the rational case. In Figure 5b, we can see that while the largest proportion behave rationally, a large portion of people fall close to the anchoring bias update (close to 0), while only a small number appear to over-update and have an update near the recency bias.

However, these observations are only for one round and a single employee. To get a better estimate for what kind of bias each employee might be subject to, we should analyze the trend over all of the available rounds. Thus, we can categorize these workers by taking a linear regression of the rational updates(Δp) on the manager's updates ($\Delta\tilde{p}$):

$$\Delta\tilde{p} = \alpha + \beta\Delta p$$

and can use the estimated slope (β) as a quantifier of the biases for each participant.

Note the expected slopes for each of the following bias models:

- Complete anchoring bias: $\beta = 0$
- Rational: $\beta = 1$
- Complete recency bias: $\beta = 2$

4.2 Analyses

Running the bias estimation previously described, we obtain the distribution of slope seen in Figure 6.

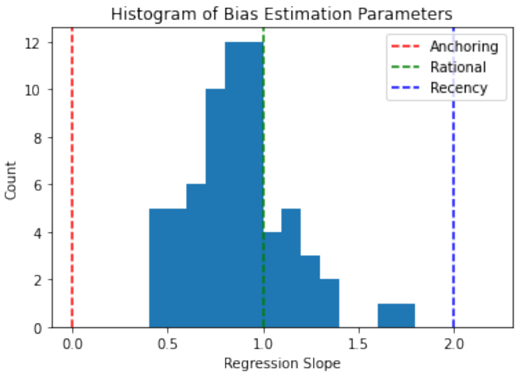


Fig. 6. Distribution of regression slopes, which are used as a parameter for biases, removing regressions which have $R^2 < 0.1$.

As a result of the regressions, we can also see that the majority of intercepts lie between $[-0.05, 0.05]$ which is expected. Additionally, we remove any entries with $R^2 < 0.1$ as those cases have too much noise to meaningfully extrapolate any information from. This removes 15 participants and still leaves us with 71 slopes.

Similarly to the distribution over a single round and worker, most of the participants have a slope close to 1, matching the rational updating rule. We do again see a portion of participants below the rational updating rule, and a very small number above. We can observe a marked decrease in the magnitude of the deviations from the rational case, as neither the anchoring bias nor the recency bias slopes are near any remaining observations. Although this can be attributed to a higher degree of noise and limitations regarding testing of biases. We are implicitly testing for biases within the 5 day period timeframe, however this may not be entirely accurate for different people. But, overall this does provide evidence for the existence of these biases, justifying our assumptions and indicate some greater tendency towards anchoring biases for human managers.

Regarding the hypotheses we wanted to test, **H1–H3** all gave insignificant results. For **H4**, we test it using a question in the post-game questionnaire that polled participants on how much emphasis they placed on getting high performance versus learning worker's preferences. The question had them rate their placement of importance on a scale from 1 to 7.

We can observe that participants with small reward low preference score condition tended to place much more emphasize on high performance, disproportionately rated a 6 or 7, while the small penalty condition saw more conservative ratings between 3 to 5. Running a chi-squared goodness of fit test, we can see that this distribution difference is indeed statistically significant, with a χ-squared value of 17.561, df of 6, and a p-value of 0.007428. Thus, we can conclude low reward participants had a different distribution, which tended to focus on high performance. Overall, from this section we can observe a plausible distribution of biases of real human managers and additionally, confirm how strategy regarding prioritization of performance or preference can depend on whether low scored are penalized.

5 Discussion

5.1 Model Applications

Now that we have established evidence for the existent of and the types of biases present in human managers, we can incorporate this into our existing theoretical model in order to evaluate the difference effects of human manager and AI managers on workers.

The main change that we make to the model is in the manager's estimate of p, $\hat{p}$. The updated estimate equations are listed below for a bias duration of k periods:

- Complete recency bias:

$$\hat{p}_i = \frac{1 + \sum_{t=\max\{1,n-k+1\}}^{n} X_{it}}{2 + \min\{n, k\}} \tag{6}$$

- Complete anchoring bias:

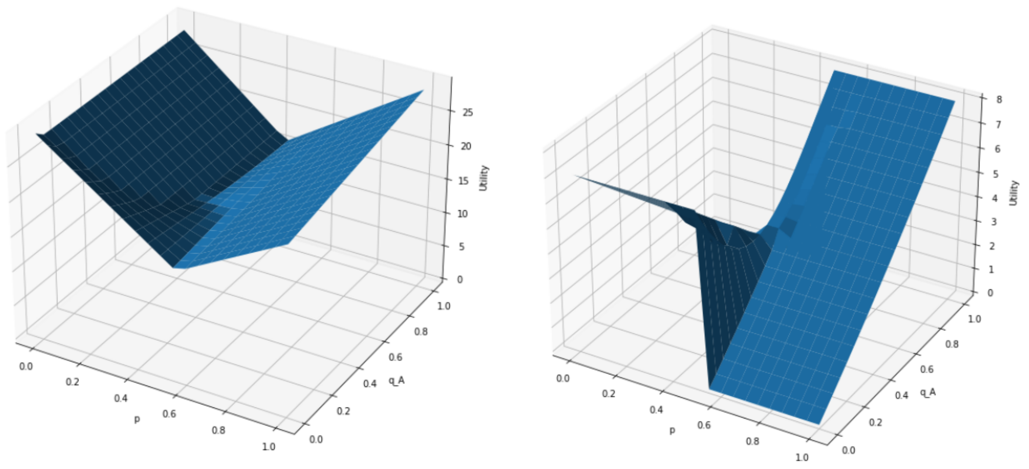
$$\hat{p}_i = \frac{1 + \sum_{t=1}^{\min\{n,k\}} X_{it}}{2 + \min\{n, k\}} \tag{7}$$

and are also applicable for q_{Ai} and q_{Bi} estimates.

Additionally, we can now relax the **assumption of fixed worker performance**, allowing workers to potentially improve and signal investment in certain types of jobs. This is achieved by having the algorithmic manager infer q_A and q_B with recency bias in mind, adapting to the constraint of shifting performance. We incorporate this adjustment of the algorithm in all future results.

We can take equations (6) and (7), and plug them into the model developed in Section 2. However, due to the nature of these biases, the results are uninteresting and trivial for the two period case which we previously derived. So we want to consider this model for a higher number of periods, which will require computer assistance and simulation due to the exponential growth of sub-game we must consider and the piece-wise nature of these utilities and functions.

It is helpful to establish a few mathematical expressions regarding our derivation of the expected utility calculations. In all of these, the $\hat{p}$ estimate will get updated by whatever bias that the worker believes the manager is operating under.



(a) Worker utility under a rational manager.

(b) Manager utility under a rational manager.

Fig. 7. Simulation Utility Graphs for Managers and Workers under a Rational Manager for $n = 7$ periods

For notation, let

$$A_{it} = \begin{cases} 1 & \text{if worker i is assigned job A at time period t} \\ 0 & \text{otherwise} \end{cases}$$

which is a function of the manager's estimates detailed in equation (3).

Assume that the worker is operating on a finite time horizon of n, then we can say that the future expected value at time t can be represented by:

- The last period n will still see the worker accepting whatever they are assigned:
$$\mathbb{E}_{in}[U|A_{in}] = u_2(pA_{in} + (1-p)(1-A_{in})) + u_1((1-p)A_{in} + p(1-A_{in}))$$
- The following is a recursive relationship that relies on expected utility in the next time period. Note that all of the A_{it+1} are dependent on how the estimates get updated based on the observations in the most recent period:
$$\begin{aligned} \mathbb{E}_{it}[U|A_{it}] = \max\{ & u_2(pA_{in} + (1-p)(1-A_{in})) + u_1((1-p)A_{in} + p(1-A_{in})) \\ & + \delta(q\mathbb{E}_{it+1}[U|A_{it+1}, Y_{it} = 1] + (1-q)\mathbb{E}_{it+1}[U|A_{it+1}, Y_{it} = 0]), \\ & \delta\mathbb{E}_{it+1}[U|A_{it+1}, Z_{it} = 0]\} \end{aligned}$$
for any $t \in [2, n-1]$. The case of $t = 1$ is the same as above, except all δ are replaced with $\beta\delta$ terms.

We can first see the direct results over the p-q_A parameter space (here we assume $q_B = 1 - q_A$ for simplicity), for the following fixed values $u_H = 1$, $u_L = 0$, $u_2 = 5$, $u_1 = 1$, and $\delta = 0.9$ assuming the manager is rational.

The following figures list the rejection region over where the worker will reject the first assigned task, and the expected worker and manager utilities over those regions.

We can see some interesting details, specifically how workers benefit when their preferences are at either extreme and how managers benefit when the values of p and q_A "match" each other, i.e., workers prefer types of jobs that they also happen to be good at.

Now incorporating the biases, we can compare the differences that we see. For these graphs in Figure 7, we change the fixed parameters to $u_H = 1$, $u_L = -1.5$, $u_2 = 1.5$, $u_1 = 1$, and $\delta = 0.9$. We also consider this under the condition of $k = 2$ for both the anchoring and recency biases. You can picture this as a two-period window of observations we take into consideration that either fixed at the start in the anchoring bias case, or is moving at the most recent period in the recency bias case.

This gives us the following rejection regions, as well as the following differences in expected utilities for managers and workers shown in Figure 8.

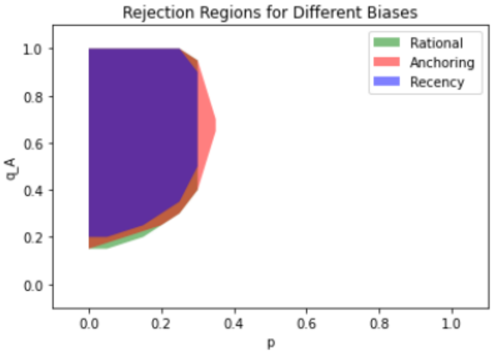


Fig. 8. Different rejection regions for rational and the two biases, anchor and recency.

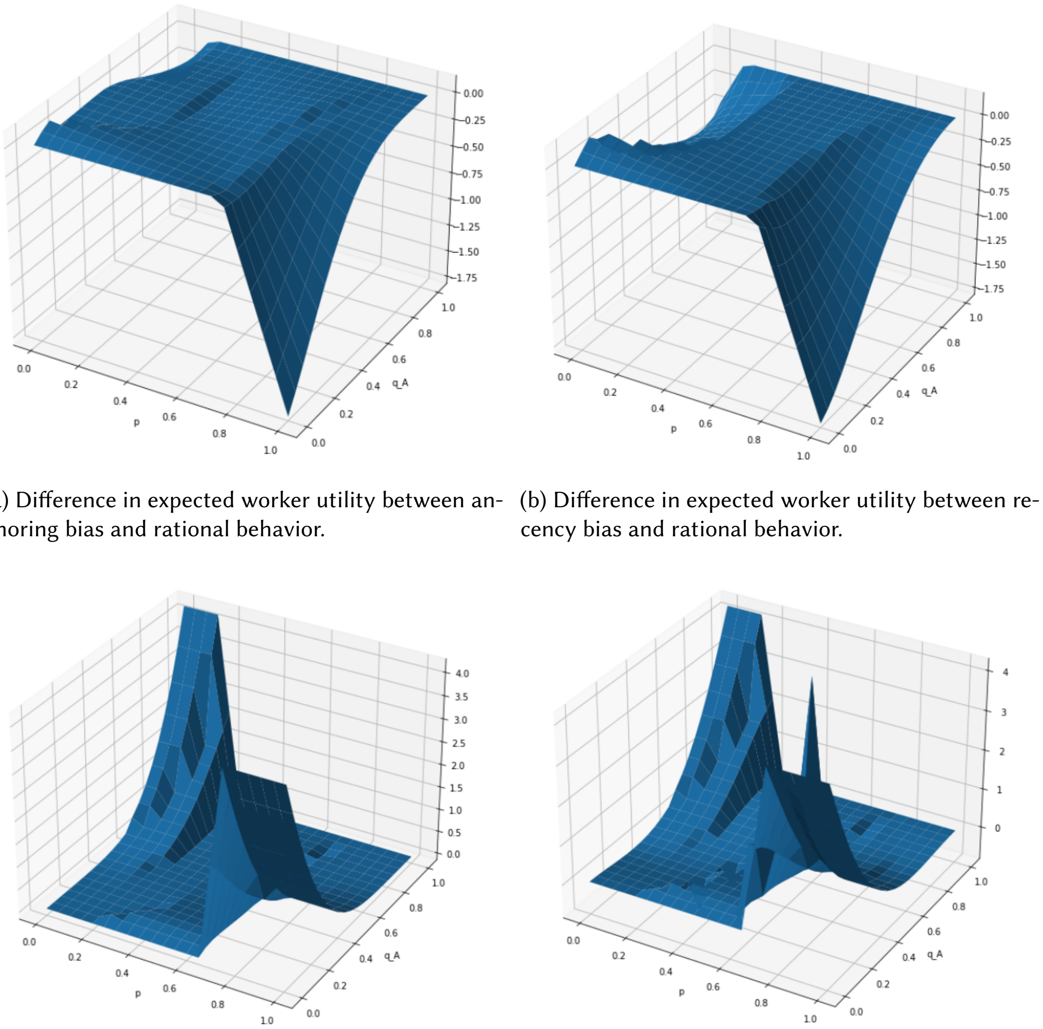
Looking at the rejection plot region, we find that the anchoring bias has the largest rejection region, with the rational and recency bias regions having slightly different but comparable regions. This makes fairly intuitive sense as a worker who thinks their manager only values the first two observations, will be much more picky about signaling their preferences in that case.

The plots reveal several notable insights, but the most striking observation is the significant decrease in worker expected utility when the values of p and q_A diverge. However, over that same region, we observe a substantial increase in manager utility. One can rationalize this as manager biases preventing some of the worker "gaming" and signaling that occurs in these regions. As these biases make managers more difficult to influence, we see manager utility increase as workers become more likely to accept jobs they dislike, but are very good at (definition of this region). Consequently, this is why we see the decrease in worker utility in those same regions.

Overall, these graphs showcase a particular example where human managerial biases can actually benefit the platform/manager side at the expense of worker utility. Although it's important to note that we see these results for these fixed values specifically, so more work needs to be done to analyze how these results how up over a variety of different parameters.

5.2 Implications for Platforms under Algorithmic Management

From these results we observe how under the game theoretic model, workers' actions in anticipation of manager changing their preference and performance estimates, thus potentially changing future assignment produce these surprising results. Here, the assumed perfect memories of algorithmic management systems make workers more likely to try and manipulate these estimates to their advantage. However an assumed imperfect memory of a human manager creates a situation where they are less susceptible to this manipulation, thus reducing worker "gaming". This effect becomes even more compounded if we consider more positive worker attitudes towards human managers documented in the literature, which may additionally make workers more reluctant to try and game human managers as comparing to AI [Lee, 2018].



(a) Difference in expected worker utility between anchoring bias and rational behavior.

(b) Difference in expected worker utility between recency bias and rational behavior.

(c) Difference in expected manager utility between anchoring bias and rational behavior.

(d) Difference in expected manager utility between recency bias and rational behavior.

Fig. 9. Differences in manager and worker expected utilities in simulations of recency and anchoring biases.

Within this context, these differences really only become noticeable in the cases of worker performance and preference clashing. To frame it within the tension between autonomy and control described in the literature, the worker can be thought of only caring about their preferences, p, in the maximization problem, while managers must juggle both preferences and performances in their consideration, but particularly care about performance, q_A, in the case of penalties for low performance, $u_L < 0$. When worker's performance abilities roughly match their preferences ($p \approx q_A$), workers and managers are aligned in their objective function and the game takes a more collaborative setting. Thus the previously described "gaming" that is more likely under AI, doesn't have much reason to occur here. This is reflected in an insignificant utility difference for either managers or workers in these regions, shown in Figures 9a–9d.

However, when we are in the regime of conflict preference and performance conditions ($p \neq q_A$), then this difference in worker "gaming" and cherry-picking re-emerged and is reflected by large disparities in both worker and manager utility between human and AI managers for both biases. Under these conditions, worker's and manager's objective become more adversarial. So when introducing human managers, which remove some of the power that workers have to influence managers, some of that adversarial framework is reduced and the system return somewhat to that previous collaborative setting.

The benefits of this arrangement are clear for the managers who sees large increases in utility under human biases compared to AI managers (Figures 9c, 9d). On the other hand, this arrangement initially appears to be negative from a worker perspective, who sees similar magnitude decreases in expected utility over these regions under human biases (Figures 9a, 9b). However, these measures of utility are purely from a worker's preference perspective. From a more utilitarian point of view, under human biases, the worker is more likely to accept recommended jobs rather than rejecting to signal. This results in potentially more earned revenue for the worker as they completed more assignments, albeit at a loss to their autonomy. Although, this decrease in perceived autonomy may still be beneficial in cases where worker perceive autonomy is higher than reality, examples include restrictions on types of jobs due to variable demand.

Ultimately, the presence of these human biases serve to ensure jobs are more efficiency allocated to the corresponding high performance workers and align worker and platform objectives to increase apparent collaboration. Platforms may want to investigate feasibility of integrating some of these biases into their systems, whether it be by integrating human managers into the algorithmic workflow or by some other means of signaling.

5.3 Extensions

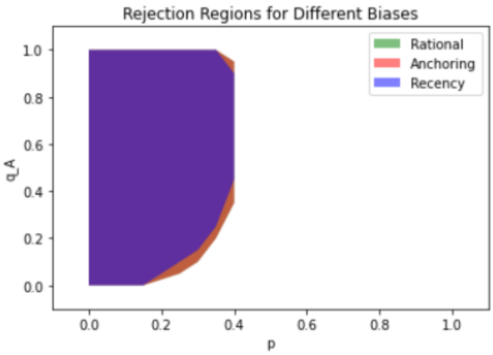


Fig. 10. Different rejection regions for rational and two biases, where time for rejection and acceptance differ.

5.3.1 Different Time Durations. A small robustness check can be done by way of relaxing the assumption that rejection of a task will take the same amount of time for a worker as the acceptance and completion of a task (both taking one period of time). Supposing that rejecting a task takes one time period, while accepting a task takes d time periods, then our previous recursive relation for a worker's expected utility now becomes:

$$\mathbb{E}_{it}[U|A_{it}] = \max\{u_2(pA_{it} + (1-p)(1-A_{it})) + u_1((1-p)A_{it} + p(1-A_{it}))$$

$$+\delta(q\mathbb{E}_{it+d}[U|A_{it+d}, Y_{it} = 1] + (1 - q)\mathbb{E}_{it+d}[U|A_{it+d}, Y_{it} = 0]),$$

$$\delta\mathbb{E}_{it+1}[U|A_{it+1}, Z_{it} = 0]\}$$

Similarly, the formulas for the manager estimates of p, q_A, and q_B would be updated to include time periods with acceptances or rejections as opposed to all time periods.

Running code simulations over almost the same parameters: $u_H = 1, u_L = -0.1, u_2 = 1.5, \delta = 0.9$ as before, but now we increase the number of time periods to 12 with acceptances taking up $d = 2$ time periods.

With the relaxation of the time assumption, figure 10 shows the rejection region again for the first time period. Here we actually see a much larger rejection region for all three type of managers than we did earlier due to a decreased penalty for rejected an assigned job. Although, the differences in the rejection regions do become less apparent with this relaxation, as anchoring bias' region is only slightly bigger than the other two.

Figure 11 is a similarly representation of the differences in worker and manager utilities to figure 9, but with a acceptance/rejection time duration difference. We see a very similar effect to the simulation without the relaxation, where regions of conflicting preferences and performance see decreases in worker expected utility and increases in manager utility. In fact, this effect is somewhat magnified as it is seen over a slightly increased parameter region (the border without the relaxation is above $p = 0.6$ while with the relaxation is around $p = 0.6$).

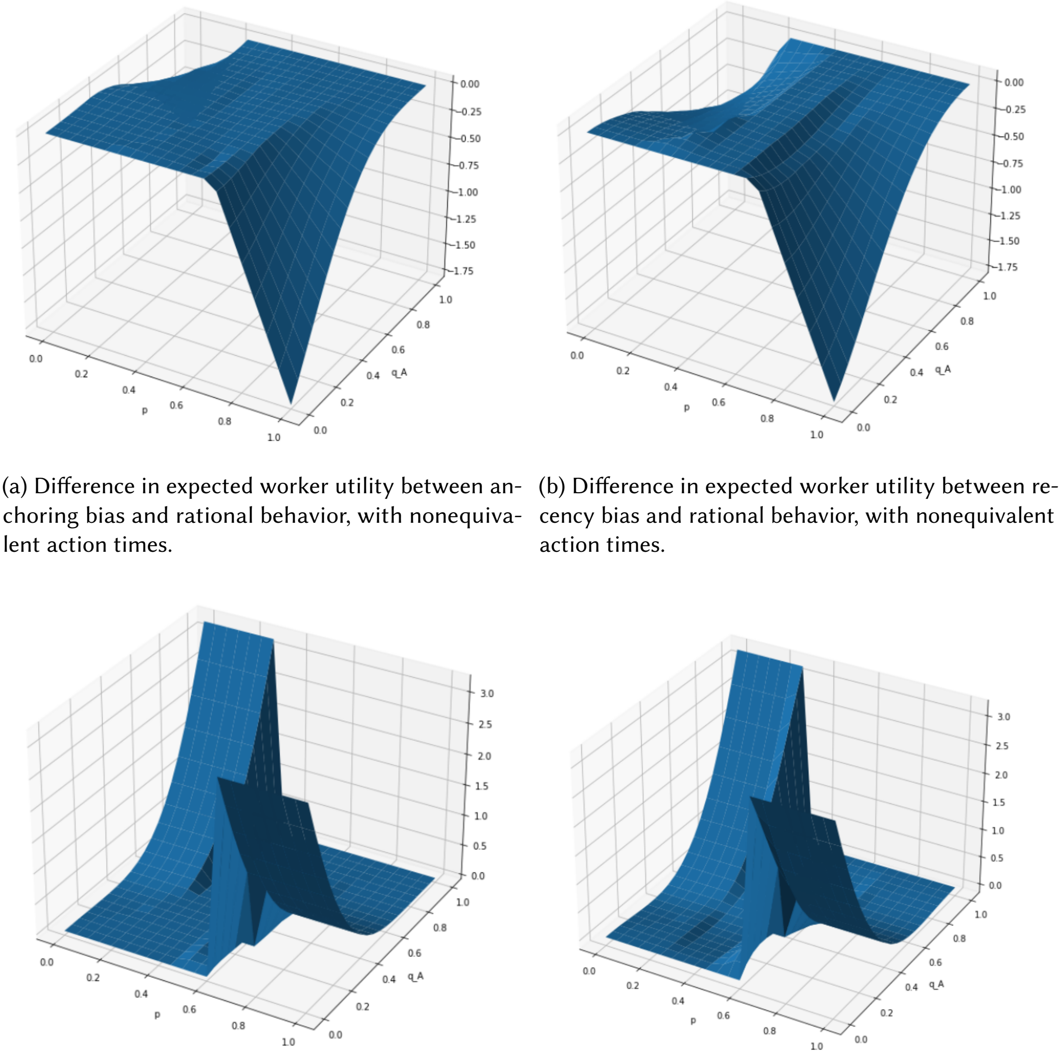
6 Concluding Remarks

This study highlights the complexities of integrating AI-driven management systems into workplace environments, particularly in task allocation. Our findings show that human biases, such as recency and anchoring effects, significantly influence managerial decision-making, sometimes leading to beneficial outcomes but often causing inefficiencies. We have specifically detailed conditions and regions where these biases benefit platforms at some cost to workers. AI systems, while capable of mitigating these biases through data-driven approaches, must also account for the nuanced interactions between workers and AI decision-makers. The tendency of workers to adapt strategically, resist, or “game" the system complicates the straightforward replacement of human managers with AI.

Further extensions of this work would help to further flesh out the simple model and reduce the number of assumptions that would need to be made. Specifically, introducing the idea of a limited supply with arrival rates and multiple workers would potentially lead to some interesting conclusions under a wider array of circumstances. Additional extensions such as workers incorrectly predicting biases and difference in job pricing would also help to deepen our current findings. Further experimentation more on the worker side would also prove benefit in further demonstrating the validity of this framework.

Importantly, our results demonstrate that these dynamics are not limited to task allocation alone. The broader implications extend to various managerial functions where AI systems are introduced, such as performance evaluation, resource distribution, and strategic decision-making. The interaction between AI-driven processes and human behavior suggests that AI tools need to be adaptable to context-specific factors and human values like fairness, transparency, and worker engagement. These considerations are crucial for fostering acceptance of AI management and realizing the full potential of AI to enhance productivity without sacrificing the social dynamics of workplace environments.

As AI continues to permeate managerial roles, the key challenge will be designing systems that not only optimize efficiency but also align with human behaviors and expectations. Our research provides a framework for understanding the trade-offs between human and AI management–and



(a) Difference in expected worker utility between anchoring bias and rational behavior, with nonequivalent action times.

(b) Difference in expected worker utility between recency bias and rational behavior, with nonequivalent action times.

(c) Difference in expected manager utility between anchoring bias and rational behavior, with nonequivalent action times.

(d) Difference in expected manager utility between recency bias and rational behavior, with nonequivalent action times.

Fig. 11. Differences in manager/worker expected utilities with different acceptance/rejection time durations.

even human-in-the-loop–offering insights that can be generalized to a range of decision-making contexts beyond task allocation. Future studies should focus on refining these systems to balance the adaptive capabilities of human decision-makers with the consistency and scalability of AI, ensuring they are flexible enough to address the evolving needs of modern organizations.

References

Zahra Ashktorab, Q Vera Liao, Casey Dugan, James Johnson, Qian Pan, Wei Zhang, Sadhana Kumaravel, and Murray Campbell. 2020. Human-ai collaboration in a cooperative game setting: Measuring social perception and outcomes. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW2 (2020), 1–20.

Max H Bazerman and Don A Moore. 2012. *Judgment in managerial decision making*. John Wiley & Sons.

Masha Borak. 2022. China's gig workers are challenging their algorithmic bosses. *Wired* (2022). https://www.wired.com/story/chinas-gig-workers-challenging-algorithmic-bosses

David Brooks. 2020. Amazon drivers hanging phones in trees to get more deliveries. *NYPost* (2020). https://nypost.com/2020/09/03/amazon-drivers-hanging-phones-in-trees-to-beat-competition

Eliane Léontine Bucher, Peter Kalum Schou, and Matthias Waldkirch. 2021. Pacifying the algorithm–Anticipatory compliance in the face of algorithmic management in the gig economy. *Organization* 28, 1 (2021), 44–67.

Lindsey D Cameron and Hatim Rahman. 2022. Expanding the locus of resistance: Understanding the co-constitution of control and resistance in the gig economy. *Organization Science* 33, 1 (2022), 38–58.

Craig R Carter, Lutz Kaufmann, and Alex Michel. 2007. Behavioral supply management: a taxonomy of judgment and decision-making biases. *International Journal of Physical Distribution & Logistics Management* 37, 8 (2007), 631–669.

Alessandro Delfanti. 2021. Machinic dispossession and augmented despotism: Digital work in an Amazon warehouse. *New Media & Society* 23, 1 (2021), 39–55.

Berkeley J Dietvorst, Joseph P Simmons, and Cade Massey. 2015. Algorithm aversion: people erroneously avoid algorithms after seeing them err. *Journal of experimental psychology: General* 144, 1 (2015), 114.

Elena Fumagalli, Sarah Rezaei, and Anna Salomons. 2022. OK computer: Worker perceptions of algorithmic recruitment. *Research Policy* 51, 2 (2022), 104420.

K Hao and N Freischlad. 2022. The gig workers fighting back against the algorithms. *MIT Technology Review* (2022). https://www.technologyreview.com/2022/04/21/1050381/the-gig-workers-fighting-back-against-the-algorithms

Mohammad Hossein Jarrahi, Will Sutherland, Sarah Beth Nelson, and Steve Sawyer. 2020. Platformic management, boundary resources for gig work, and worker autonomy. *Computer supported cooperative work (CSCW)* 29 (2020), 153–189.

Kalle Kusk and Midas Nouwens. 2022. Platform-mediated food delivery work: a review for CSCW. *Proceedings of the ACM on Human-Computer Interaction* 6, CSCW2 (2022), 1–25.

Min Kyung Lee. 2018. Understanding perception of algorithmic decisions: Fairness, trust, and emotion in response to algorithmic management. *Big Data & Society* 5, 1 (2018), 2053951718756684.

Min Kyung Lee, Daniel Kusbit, Evan Metsky, and Laura 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*. 1603–1612.

Xiaowen Liu and et al. 2019. Bonus competition in the gig economy. *SSRN* (2019). https://doi.org/10.2139/ssrn.3392700

Ning F Ma, Veronica A Rivera, Zheng Yao, and Dongwook Yoon. 2022. "Brush it Off": How Women Workers Manage and Cope with Bias and Harassment in Gender-agnostic Gig Platforms. In *Proceedings of the 2022 CHI conference on human factors in computing systems*. 1–13.

Yunke Mai, Bin Hu, and Saša Pekeč. 2023. Courteous or crude? Managing user conduct to improve on-demand service platform performance. *Management Science* 69, 2 (2023), 996–1016.

Mareike Möhlmann and Ola Henfridsson. 2019. What people hate about being managed by algorithms, according to a study of Uber drivers. *Harvard Business Review* 30, August (2019), 1–7.

Lillio Mok, Sasha Nanda, and Ashton Anderson. 2023. People perceive algorithmic assessments as less fair and trustworthy than identical human assessments. *Proceedings of the ACM on Human-Computer Interaction* 7, CSCW2 (2023), 1–26.

Isabel Munoz, Steve Sawyer, and Michael Dunn. 2022. New futures of work or continued marginalization? The rise of online freelance work and digital platforms. In *Proceedings of the 1st Annual Meeting of the Symposium on Human-Computer Interaction for Work*. 1–7.

Hyanghee Park, Daehwan Ahn, Kartik Hosanagar, and Joonhwan Lee. 2021. Human-AI interaction in human resource management: Understanding why employees resist algorithmic evaluation at workplaces and how to mitigate burdens. In *Proceedings of the 2021 CHI conference on human factors in computing systems*. 1–15.

Divya Ramesh, Caitlin Henning, Nel Escher, Haiyi Zhu, Min Kyung Lee, and Nikola Banovic. 2023. Ludification as a Lens for Algorithmic Management: A Case Study of Gig-Workers' Experiences of Ambiguity in Instacart Work. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference*. 638–651.

Alex Rosenblat and Luke Stark. 2016. Algorithmic labor and information asymmetries: A case study of Uber's drivers. *International Journal of Communication* 10 (2016), 3758–3784. https://ijoc.org/index.php/ijoc/article/view/6205

Amos Tversky and Daniel Kahneman. 1974. Judgment under Uncertainty: Heuristics and Biases: Biases in judgments reveal some heuristics of thinking under uncertainty. *Science* 185, 4157 (1974), 1124–1131.

Rui Zhang, Nathan J McNeese, Guo Freeman, and Geoff Musick. 2021. "An ideal human" expectations of AI teammates in human-AI teaming. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW3 (2021), 1–25.

A Proofs of Theorems

A.1 Proof of Theorem 2.1

The problem is a simple maximization problem over the space of the probability that the manager assigns a job of type A, $x \in [0, 1]$:

$$\mathbb{E}_m[U|x] = x * \mathbb{E}_m[U|\text{assign type A}] + (1 - x) * \mathbb{E}_m[U|\text{assign type B}]$$

which is simply a linear equation with respect to x. Thus, when the slope is positive, manager expected utility is maximized when $x = 1$, aka always assigning task A. When the slope is negative, the optimal value is $x = 0$, so the manager should always assign type B. When the slope is 0, any strategy is optimal, so for simplicity, we choose to assign $x = 1$ under this condition.

Then solving explicitly for this slope condition (with all worker parameters as the manager's estimated values):

$$\begin{gathered}
\mathbb{E}_m[U|\text{assign type A}] - \mathbb{E}_m[U|\text{assign type B}] \geq 0 \\
p(q_A u_H + (1 - q_A)u_L) - (1 - p)(q_B u_H + (1 - q_B)u_L) \geq 0 \\
(pq_A - q_B + pq_B)u_H + (p - pq_A - 1 + q_B + p - pq_B)u_L \geq 0 \\
(pq_A + (p - 1)q_A)(u_H - u_L) + (2p - 1)u_L \geq 0 \\
pq_A + (p - 1)q_B \geq (1 - 2p)\frac{u_L}{u_H - u_L}
\end{gathered}$$