PLOS Complex Systems

RESEARCH ARTICLE

Reinforcement learning to develop policies for fair and productive employment: A case study on wage theft within the day-laborer community

Matt Kammer-Kerwick[1*], **Evan Aldrich**[2]

1 Bureau of Business Research, IC² Institute, The University of Texas at Austin, Austin,Texas, United States of America, **2** Department of Economics, Texas A&M University, College Station, Texas, United States of America

* mattkk@ic2.utexas.edu

OPEN ACCESS

Citation: Kammer-Kerwick M, Aldrich E (2025) Reinforcement learning to develop policies for fair and productive employment: A case study on wage theft within the day-laborer community. PLOS Complex Syst 2(12): e0000079. https://doi.org/10.1371/journal.pcsy.0000079

Editor: Keith Burghardt, University of Southern California, UNITED STATES

Received: February 4, 2025

Accepted: October 15, 2025

Published: December 4, 2025

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pcsy.0000079

Abstract

This paper applies a reinforcement learning (RL) approach (batch Q-learning) to solve decision making problems toward the development of policies for fair and productive work for laborers in precarious employment situations. We present both single-agent and multi-agent settings. The first formulation more closely resembles the limited agency available to laborers today and the second is presented to address the research question of how to develop policies that allow both laborers and employers participate in employment decisions and to respond to unfair work conditions. The single agent formulation confirms a policy often observed in practice where day laborers take jobs with the risk of wage theft and endure the outcome because the likelihood of achieving justice is low and the laborer typically still receives a fraction of their wages. We demonstrate that the two-agent formulation allows the policy to encompass decisions by both laborers and employers. Within this decision-making dynamic, we illustrate through sensitivity analysis that under modest increases in the likelihood of a successful outcome of reporting, laborers learn to report theft and employers learn not to steal. We use the complexity of the case study examined to motivate a more general formulation based on the generalized semi-Markov process that allows the method to incorporate more detailed system dynamics that, in turn, allow for more precise policies to be formulated and determined. We discuss the implications of both the policies determined in the case study and the potential of the generalized semi-Markov reinforcement learning formulation.

Author summary

This study continues previous translational research at the intersection of behavioral and decision science toward the larger goal of designing dynamic operational policies for complex sociological systems. While this paper has a

Data availability statement: All data used in our models have been previously published. Our python code has been included in the supplement as a PDF file.

Funding: This work was supported by the National Science Foundation (2039983 to MKK). The funder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

methodological orientation, we situate a demonstration of our evidence-based model in the problem domain of "decent work" as envisioned by the United Nations Sustainable Development Goal 8. Here, we apply and extend reinforcement learning to observe the system, periodically try new policy ideas, and adapt the deployed policy based on feedback from the system under the new policy candidate. We apply our approach to the societal challenge of disrupting wage theft experienced by day laborers. We use this case study to recognize the benefit of adapting our approach to complex systems by incorporating a more generalized stochastic model.

1. Introduction

Precarious employment, characterized by uncertainty and risk for the worker, is a pervasive societal challenge, with wage theft representing a significant form of exploitation within this domain. Existing approaches to address such issues often struggle to account for the dynamic interactions between laborers and employers and to develop adaptive policies that effectively mitigate exploitation. This paper addresses this gap by applying a reinforcement learning (RL) approach to formulate dynamic policies for fair and productive employment, specifically focusing on wage theft within the day-laborer community. This study builds directly on our previous work. Our empirical research [1] has characterized the context of day labor, including rates and extent of exploitation, laborer decision-making under partial agency, and the potential role of worker centers. This foundational understanding and data informed our subsequent work [2], where we extended a behavioral-science based intervention-design framework by incorporating agent-based modeling (ABM) to virtually examine hypothetical interventions. The current paper advances this research by applying a model free reinforcement learning methodology to learn optimal policies within this adaptive framework, demonstrating how policy interventions can influence both laborer and employer behavior to reduce wage theft. This RL approach offers a powerful tool for developing adaptive, evidence-based policies in complex, stochastic environments.

Methodologically, we solve decision-making problems in a stochastic environment modeled as an embedded discrete Markov chain in either a single-agent or multi-agent setting. The methodology transcends traditional paradigms, opting for a model-free reinforcement learning approach [3]. Unlike conventional modeling, this approach embraces a direct learning paradigm, where the agent interacts with the environment without explicitly modeling its dynamics. This process enables the agent to glean valuable insights and directly learn either a policy or a value function from observed interactions, illustrating the adaptability and versatility of the proposed reinforcement learning framework. This study presents a generalized adaptive model-free reinforcement learning framework suitable for navigating decision-making complexities within complex sociological systems, illustrated in a case study on wage theft. It offers a generalized research process applicable to the specific problem

examined here and to a wide array of sociological research endeavors. This process is described by the authors in [2] as a decision-science extension of a behavioral-science based intervention-design framework [4].

The contributions made by this paper are summarized as follows:

1. We apply a batch Q-learning approach to model day laborer and employer decision-making regarding wage theft, demonstrating its efficacy in learning dynamic optimal policies for fair employment.
2. We present both single-agent and multi-agent RL formulations, showing how the multi-agent model can inform policies that account for co-evolving behaviors of laborers and employers.
3. We illustrate through sensitivity analysis that under modest increases in the likelihood of a successful outcome of reporting, laborers learn to report theft and employers learn not to steal.
4. We motivate and introduce a more general formulation based on the generalized semi-Markov process (GSMDP) to incorporate more detailed system dynamics and enable the formulation of more precise future policies.

The remainder of this paper is structured as follows: Section 4 provides essential background on precarious labor and exploitation, our previous empirical and ABM-based modeling of a real-world instance of that problem, and introduces the RL and stochastic modeling theory we use in this paper to develop policies for that problem domain. Section 5 develops our theoretical framework for single- and multi-agent formulations and the computational methodology used. Section 6 presents the results of our ABM-based RL exercises to demonstrate the application of our approach. Section 7 discusses the generalization of our model to a more complex framework. Finally, Section 8 provides a discussion of our findings, their implications, and directions for future research

2. Background

This section provides essential background on the behavioral-science-based intervention-design framework in [2], reviews the literature on the problem domain of precarious labor, summarizes our previous empirical and ABM-based examinations of that problem [1], and introduces the theoretical underpinnings of our stochastic modeling and RL approach used in this paper to develop policies for that problem domain.

2.1. Behavioral-science-based intervention-design framework

In a study on illegal fishing, Battista and colleagues [4] observed that insights from behavioral science have the potential to support the design of interventions for modifying specific illicit behaviors. Their framework begins with stakeholder involvement to ensure that the process has a foundation of norms, beliefs, and community thinking about the behavior targeted for intervention. In this way, their process recognizes the power of primary and even primordial prevention [5,6] to disrupt illicit behavior and promote desirable alternative behaviors. Literature and artefactual experiments aid this design process before small pilots and deployment of interventions at scale. Our study [2] extended the original process with additional steps to allow for virtual examination of hypothetical interventions in a simulation environment and machine learning methods (RL specifically) to optimize candidate policies prior to tests and deployment in pilots or scaled-up deployments. Fig 1 illustrates the extended process. It is important to note that the framework is an iterative learning process that adds data and model refinements over time to produce more complete and optimal operating policies.

This paper illustrates a practical application of the framework above by examining optimal strategies in the domain of disrupting labor exploitation. This research continues previous work by the authors [2], which focused on designing and evaluating interventions to address labor exploitation using agent-based modeling (ABM). Moreover, our study aligns with the extensive literature advocating for a well-being economy [7–9], generally, and fair employment [10–12], specifically, emphasizing the relevance and importance of our research contribution.

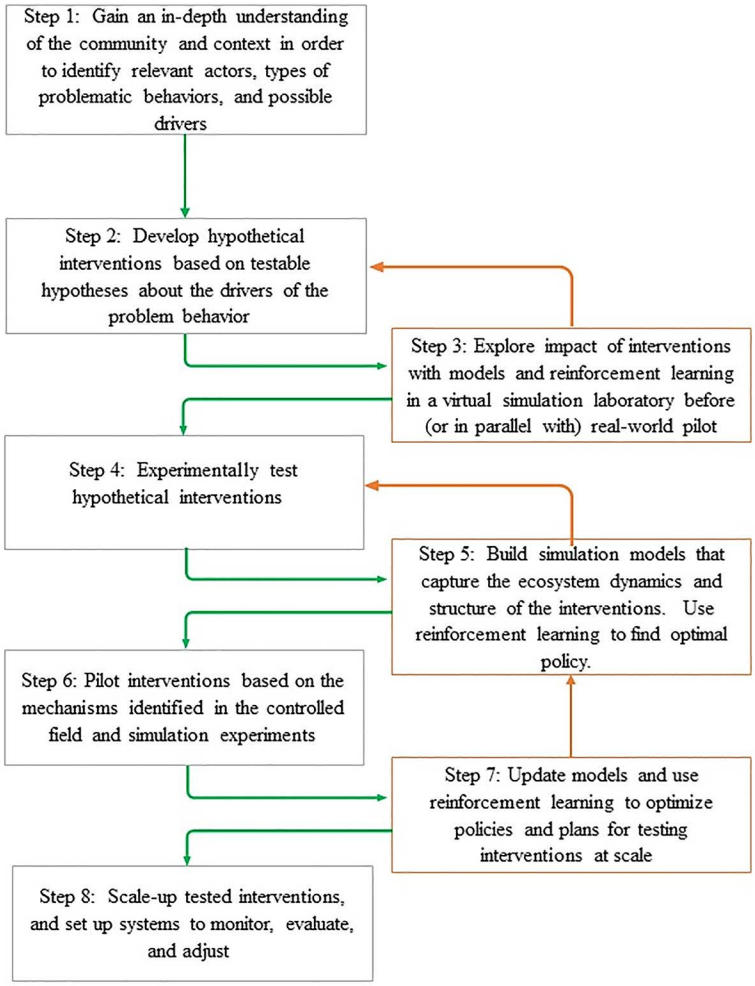

Fig 1. Proposed behavioral and decision science framework that includes artifactual data collection, synthetic data generation, and reinforcement learning to optimize the impact of interventions.

https://doi.org/10.1371/journal.pcsy.0000079.g001

Furthermore, we expand upon the proposed reinforcement learning framework to incorporate elements of dynamic adaptation, enabling the agent to adjust its strategies in response to evolving environmental conditions. By integrating adaptive mechanisms, our framework enhances its utility in dynamic and uncertain settings, presenting a robust approach for decision-making in complex socio-economic systems. This study presents an approach to optimally control complex systems that evolve in discrete time and through the occurrence of discrete events, which is well suited for reinforcement learning [13]. This approach applies to many common real-world problems where data are scarce and the agents in the system have conflicting objectives. When multiple agents apply reinforcement learning in a shared environment, the system's dynamics move beyond the traditional

Markov Decision Process (MDP) model. In such multi-agent scenarios, the optimal policy for an individual agent is not solely determined by the environment's dynamics but also depends on the policies adopted by other agents [14].

Given the specialized nature of our environment, which entails specific features and complexities like multi-agent scenarios and event spaces, we opted to construct a custom data generation environment in Python that was designed to mimic observed behaviors in a prior study of day laborers and employers. This approach allowed us to manipulate

the environment's realistic dynamics of a laborer making decisions in response to job offers, which may involve theft and reporting, prior to collecting additional data from the real world environment. The methodology utilizes Q-learning to derive efficient policies to maximize cumulative rewards over time.

2.2. Precarious labor and exploitation

We have previously examined day labor exploitation in [1,2]. Day labor is a form of precarious employment characterized by informality and, frequently, exploitation. Labor historians assert that precarious employment has been common for much of human history [15,16]. Two definitions of precarious work include Standing's precariat social category (which includes seven forms of labor security) [17] and Cranford et al.'s continuum of security (along four criteria) [18]. We adopted Kalleberg's definition [19], "employment that is uncertain, unpredictable, and risky from the point of view of the worker." Kalleberg and Hewison conceptualize precarity as a process instead of a continuum or a category, pointing to the relationships between labor, capital, and the state that produces precarious work [20]. This definition of precarity is most appropriate for our studies because it prioritizes worker perception of precarity and invites inquiry into the formal or informal character of work according to government law and its enforcement.

For our purposes, exploitation refers to a situation whereby taking advantage of another entity, the actor doing the exploitation gains more than they deserve in the interaction, and the exploited entity gets less than they deserve [21]. The distinction between the definitions of precarity, (in)formality, and exploitation is essential for our research because the day laborers we interviewed did not always perceive precarity in their informal work, even if the wage theft they experienced was exploitative, because they viewed it in the context of their migration histories. Such adaptive and vicious cycles are typical in many corrupt and exploitative settings [22–27].

Germane to our case study, laws that define formal or informal labor and the likelihood that they would be enforced are of special concern for the immigrant laborers that we have interviewed. Informal labor describes jobs that do not offer standard terms, conditions, and benefits according to state law because the law does not pertain to these jobs or because the law is not enforced [15]. A review of federal and state laws to address wage theft in the US has shown that federal enforcement is weak and state laws are highly varied [28] even though the Fair Labor Standards Act has been in effect since 1938 [29]. Our interviews [1] revealed that day laborers rarely pursue formal remedies when wage theft is experienced because of perceptions of low success rates after excessive investments of time. It is in this context that we examined how worker centers might play a convening and coordinating role in precarious employment. In fact, worker centers have evolved from formal, community-based, and community-run organizations that provide services, education, and advocacy to laborers [30] to become a response to exploitation [31].

2.3. Application domain for RL: Exploitation in day-laborer community

This study is situated within our ongoing research on precarious labor and complements our previous work which empirically explored day labor dynamics and developed an agent-based model to assess interventions. Specifically, in [1], we reported on three empirical exploratory studies among day laborers in Texas that characterized the context of the day labor experience relative to rates and extent of exploitation, the alternatives available to laborers when they are making decisions to accept work and respond to exploitative outcomes during work, and the potential of worker centers to provide an informal channel for providing educational interventions to both employers and laborers that have the potential to reduce exploitation and provide a means to respond to it when exploitation occurs. In [2], we constructed an ABM informed by the reviewed theories and the artifactual evidence from our three studies in [1]. That simulation environment allowed us to examine select policy changes and assess the sensitivity of various assumptions on the efficacy of those proposals. We concluded [2] with the observation that education for laborers and employers about workers' rights and the

obligations of employers to workers is a critically needed intervention focused on principles of labor organization optimized for the informal setting of day labor. In addition, such training should also cover workplace safety, the second most common form of labor exploitation.

These works inform the current study, which applies the RL method to illustrate how such a model free machine learning approach can observe a partially observable, stochastic system and iterate toward an optimal policy. Once again, we focus on the most common form of exploitation experienced by day laborers: wage theft. Of interest in the current study is the decision to steal is made unilaterally by the employer, typically after the laborer has accepted and begun work. Further, wage theft occurs in a fraction of jobs, and the degree of theft is typically only a fraction of the agreed-upon wages. Fig 2 illustrates the decision process of a prospective day laborer. The black lines represent action-triggered state changes, and the black circled states represent the observed state space. The day laborer begins in an idle state and remains idle until he or she is offered a position by an employer. The laborer then chooses either to accept the position or decline the position. If they accept, they enter the working state. If theft does not occur, they return to idle and have been observed as working fairly. If a theft occurs before the job ends, they enter the observed working theft state and then decide whether to report it. Regardless of the reporting decision, they return to idle with a successful report, increasing their payoff.

The most crucial piece of this decision process that this study is concerned with is the decision of the day laborer to report. In other words, is the expected payoff of putting in a report great enough to constitute going through the reporting process? And ultimately, what can be done to dissuade employers from stealing wages from laborers they hire? We begin with a single-agent version of the problem that examines the decision-making by a laborer before expanding to a multi-agent formulation that encompasses decisions by both the laborer and the employer.

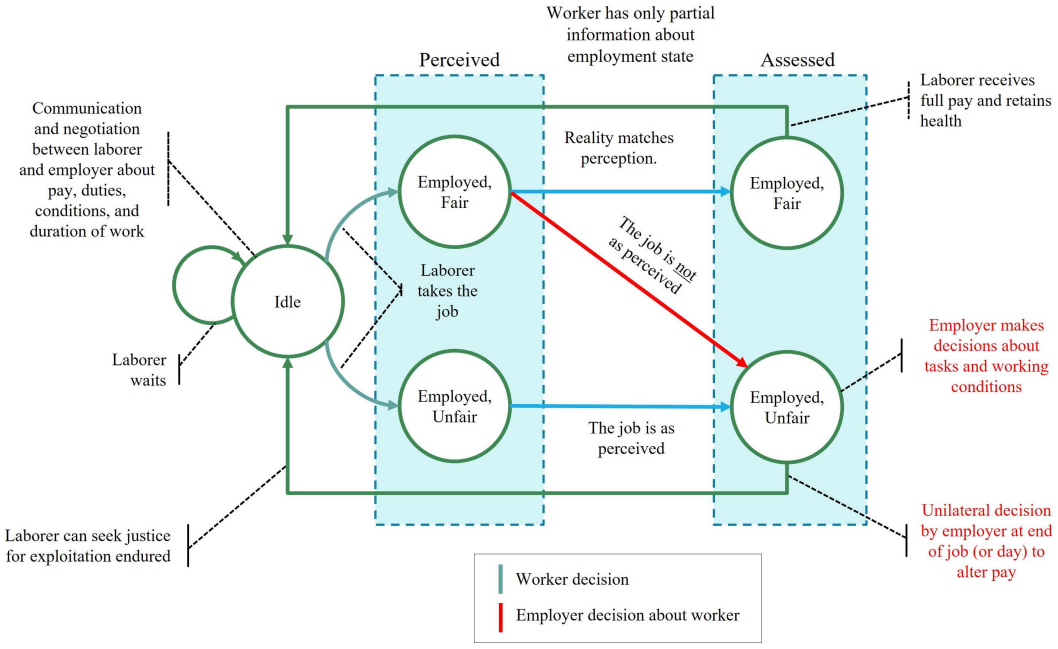

Fig 2. Employment journey of a day laborer. The employment journey of a day laborer as they accept work offered by an employer, move into a working state (fair or unfair) that is partially controlled unilaterally by that employer. Adapted from [2].

https://doi.org/10.1371/journal.pcsy.0000079.g002

2.4. Stochastic models and reinforcement learning

In complex sociological systems, stochastic models play a pivotal role in facilitating decision-making processes, particularly in scenarios where outcomes exhibit a blend of randomness and controllability by decision-makers. Among the most prevalent stochastic models employed for such purposes is the Markov Decision Process (MDP) [32]. At its core, an MDP consists of a set of states, actions, transition probabilities, and rewards, encapsulating the dynamics of sequential decision-making under uncertainty modeling a single agent. Reinforcement learning (RL) algorithms seamlessly integrate with MDPs, leveraging their structure to learn optimal decision-making policies.

Definition 1: The Markov Decision Process in the single-agent notion is defined by the following set of tuples [33]: {Sn, A_1,..., A_m, r_1,...,r_m, S_{n+1}} where:

- S is the state of the environment
- $A_{i,s}$, $i \in 1,...,m$: denotes the i^{th} actions associated with state S.
- $r_{i,s}$, $i \in 1,...,m$: denotes the reward for the i^{th} action associated with state S.
- S_{n+1} denotes the next state for the environment.

In this study, we wish to observe the driving factors for why a community member may choose to report wage theft or not based on decisions and interactions with other agents. It is reasonable to suppose that the goals of the employer and the day laborer would conflict with each other. We suspect that using Stackelberg games or extensive-form games in non-cooperative game theory may be an effective way to analyze this dynamic [34]. Additionally, efforts have been made to integrate game theory into reinforcement learning, as evidenced by these studies [14,33,35–39]. Moreover, the Markov decision process has also been widely introduced in the field of game theory with L.S. Shapley's seminal work on Stochastic Games [40,41] and further integrated with more recent studies [42–46]. We develop an extension to this literature in the methods section using an extensive form game modeled as a Markov Decision Process. We provide the formal definition of an extensive-form game from [47] here.

Definition 2: An extensive form game is a list $\Gamma = (N, V, E, x^o, (V_i)_{i \in N}, O, u)$ where

- N is the number of players.
- V is a finite set whose elements are vertices of the game tree.
- $E \subseteq V \times V$ is a finite set of pairs of vertices and denotes the edges between the vertices.
- $x^o \in V$ is the root of the game tree.
- $(V_i)_{i \in N}$ is a partition of the vertices that denotes which agent makes decisions at each vertex.

In the context of this paper, the set of vertices will denote the state space in the MDP framework, the edges represent the respective actions in each state, and a reward is assigned to both players in each state transition. In other words, during the reinforcement learning process, the agent that gets to decide at a particular vertex V observes the current state and the set of all possible future rewards. That agent then chooses their action, which moves both players to the next state. The next player then observes the new state and makes their decision.

RL enhances traditional simulation methods in multi- and single-agent contexts, providing a powerful tool for problem-solving in dynamic or uncertain scenarios [32]. RL leverages simulation models as interactive learning environments, fostering collaboration between the model and the agent to facilitate adaptive decision-making in evolving contexts. At the heart of RL lies the concept of learning from feedback. Reinforcement learning agents navigate an environment, perform an action, and receive feedback through rewards or penalties. This collaborative process uses reward and punishment mechanisms as feedback to guide agents' behavior toward developing an optimal future policy of actions.

The key components of RL include the agent, environment, actions, states, and rewards, which collectively shape the learning process.

Q-learning [48] is a widely used reinforcement learning algorithm to train agents in decision-making within environments. Q-learning operates as a model-free algorithm. The agent interacts with the environment, deriving insights from the outcomes of its actions without constructing an internal model or a Markov Decision Process. Initially, the agent knows the potential states and actions within the environment. Subsequently, the agent uncovers the state transitions and associated rewards through exploration. The Q-value, a key metric in Q-learning, quantifies the expected cumulative return or utility of taking action'a' in state's'. This value encapsulates the agent's understanding of the desirability of different actions in various states. The Q-learning update equation, represented as

$$Q(s,a) = Q(s,a) + \alpha(r + \gamma * max(Q(s',a')) - Q(s,a)) \quad (1)$$

encapsulates the iterative learning process, where α ($0 < \alpha \leq 1$) denotes the learning rate, r signifies the immediate reward, γ represents the discount factor, and $max(Q(s',a'))$ reflects the maximum Q-value for the next state-action pair. In the single-agent case, the following action is chosen by the same agent; in the multi-agent problem, the following action is selected by another agent. The learning rate determines how much the Q-values are updated in each iteration and controls the weight given to new information versus old information when updating the Q-values. On the other hand, the discount factor determines the importance of future rewards and discounts the value of future rewards compared to immediate rewards. A value closer to 1 means future rewards are more equally valued as immediate rewards, while a value closer to 0 means more immediate rewards are valued. The agent refines its understanding of optimal actions in different states by iteratively updating Q-values based on observed rewards and future estimates. As the RL algorithm progresses and the number of samples increases, Q-values converge towards more accurate estimations, facilitating enhanced decision-making and policy optimization by the RL agent.

Reinforcement learning aims to train the agent's behavior under varying circumstances, typically through trial-and-error exploration. Model-free algorithms, such as Q-learning, tackle the exploration-exploitation trade-off by balancing exploiting known actions for immediate rewards and exploring new actions for potential long-term benefit. In the epsilon-greedy method, the agent combines the exploitation of known actions with occasional exploration of new options to maintain a balance between exploiting prior knowledge and discovering new possibilities. This approach aims to maximize rewards while allowing room for experimentation. Another hyperparameter often controls this balance, typically denoted as ε in epsilon-greedy exploration strategies [49]. Reinforcement learning algorithms, particularly Q-learning, guide decision-making processes in various domains. At the core of RL lies the utility function, which quantifies the desirability of different outcomes and guides the agent towards optimal actions. Traditionally, utility functions in RL have primarily focused on monetary returns as the sole metric for evaluating outcomes. However, in many real-world scenarios, decision-making is influenced by many objectives that extend beyond mere financial gains, an area we return to in our discussion.

As a data-driven learning process, Q-learning explores the system as represented by observation. Within the decision-science extension of a behavioral-science-based intervention-design framework described above, an iterative process, training data can be sampled directly or generated from simulations built from artifactual or scaled-up empirical experiments. Fig 3 illustrates that the environment may encompass various data generation strategies, including the simplified simulation utilized in this study, a more realistic ABM (see [2]), or the real-world context (as studied in [1]). The learned policies can be tested within the simulation to evaluate their effectiveness before real-world deployment. The simulation environment serves as the training setting. The ABM can be adjusted to delegate specific decision points to the learning agent.

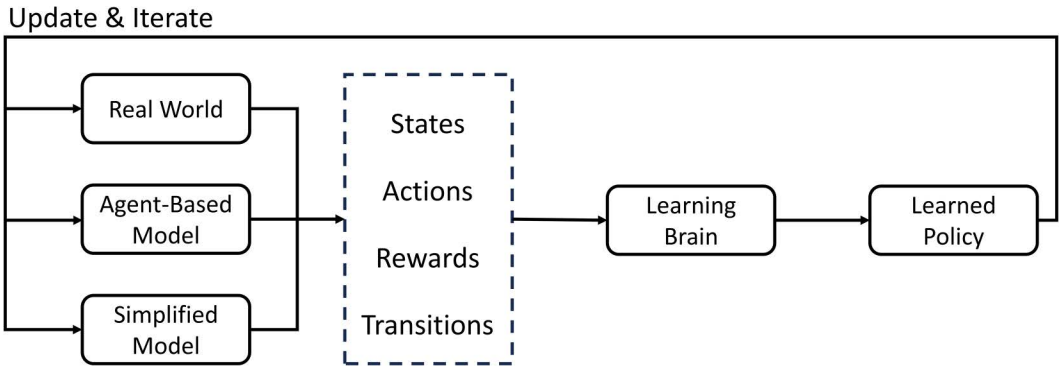

Fig 3. The batch Q-learning environment. The batch QL environment may encompass various data generation strategies, including the simplified simulation utilized in this study as a proof of concept, a more realistic ABM (see [2] for a relevant example that we used in that study of precarious labor and wage theft), or the real-world context (as examined by us in [1]).

https://doi.org/10.1371/journal.pcsy.0000079.g003

3. Methodology: Theoretical model and computational framework

Our RL algorithm takes in a dataset from a real-world environment, an agent-based modeling software, or some other data creation environment. The data is read in row by row, taking the first element as the initial state, the next element being the action taken, the reward of that path, and the subsequent resulting state. This reward is in terms of the wage the laborer incurred during their time working. In our attached supplemental document, the data creation environment is an independent function of the RL algorithm. The RL algorithm is model-free and does not take in any additional information about the environment other than the data and knowing the structure of the data being read as described above.

3.1. Stochastic models and game theoretic formulation

We present two stochastic game-theoretic formulations of our day labor decision-making problem to illustrate the flexibility of our model-free RL approach to computationally developing policies.

3.1.1. Single-agent formulation. With our application, we refer to the developments in decision theory under uncertainty. In our context, the prospective day laborer acts as a decision maker (DM) who starts in an 'idle' state and then, if offered a job, chooses whether to accept it. If accepted, the DM can decide to report the employer if they experienced wage theft. In this formulation, the theft and degree of theft are random outcomes experienced by the laborer, and the policy formulation goal is to manage those uncertainties through decisions to accept or reject jobs that are offered and to report or not when theft occurs. See Fig 4.

We characterize our decision problem as outlined below.

- The set of possible observed future states Ω = {idle, working, working theft}.
- The set of all alternative actions A = {no action, decline, accept, report}
- The set of possible consequences X = {cost of not working, $x(a;\omega)$}
- Preference relation $\succsim$ over the set of probability distributions over X.

Here, $x(a;\omega)$ is a random variable denoting, in monetary terms, the payout to the DM once they work at the job and take action a. This would include the pay for their work accounting for any stolen wages and the potential pain and suffering they gain for a successful report of wage theft. As our consequences are expressed in monetary terms the outcomes are easily comparable assuming that the same day-laborer would prefer higher pay for the same work and would strictly

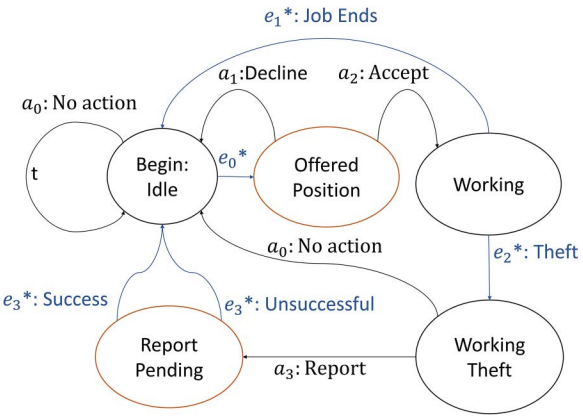

Fig 4. State/action diagram for the single agent formulation. The state, action, and rewards for the single agent formulation, including events that precipitate state changes.

https://doi.org/10.1371/journal.pcsy.0000079.g004

prefer some payment to no payment at all. Therefore, our DM would prefer action a = 'report' over action b = 'no action' if the expected payout after reporting is at least as good as the expected payout of not reporting. Or in our notation, $a \succsim b \Leftrightarrow E[x(a;\omega)] \geq E[x(b;\omega)]$. Whether to report or not is the highlight of what we are investigating.

3.1.2. Multi-agent formulation. This study also models the same situation as a multi-agent process using an extensive-form game tree that is also traversed with our reinforcement learning algorithm; see Fig 5. In this context, the following extensive-form game is defined. $\Gamma = (N, V, E, x^o, (V)_{i,i\in N}, O, u)$ where:

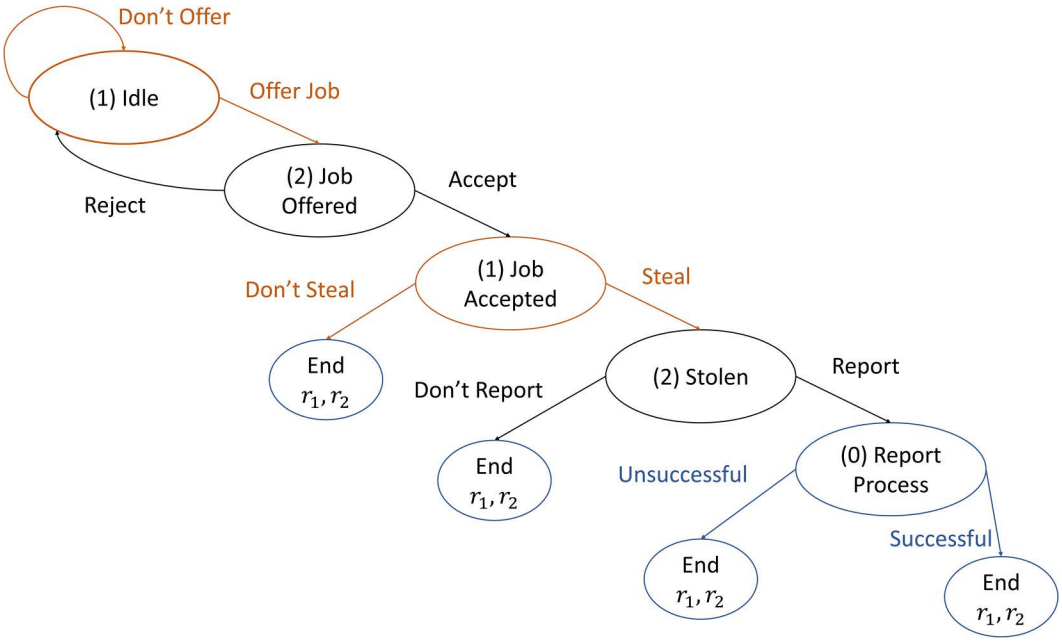

Fig 5. Game tree of the 2-player state space. In the multiagent formulation, orange denotes the states where the employer makes the decisions and the actions that can be made, black lines represent the states where the day-laborer makes decisions and the actions they can take, and blue denotes the environment and the end results. The game is immediately replayed if the job does not begin and rewards are changed for both players in each state transition.

https://doi.org/10.1371/journal.pcsy.0000079.g005

- $N = \{0,1,2\}$ where player 1 is the employer, player 2 is the day-laborer, and player 0 is the environment that does not maximize any utility.
- V = {Idle, Job Offered, Job Accepted, Stolen, Report Process, End} denoting the macrostates.
- $E \subseteq V \times V$ denotes the actions that the agents can make at each state that trigger the next state change.
- x^o is the idle state where the game begins.
- (V_1) = {Idle, Job Accepted}, (V_2) = {Job Offered, Stolen}, (V_0) = {Report Process} denotes which agent makes decisions at each vertex and "end" is the terminal node where the game ends.

3.2. Computational framework

We present the ABM-based data generation approach used and the model free RL Framework used to demonstrate policy development for the problem domain of precarious labor and interventions to address wage theft in that environment.

3.2.1. Data Generation via agent-based modeling environment. We have implemented ABM-based environments to generate data for both the single-agent and multiagent formulations to illustrate how the RL methodology we are demonstrating can adapt to a deeper understanding of the problem domain through our iterative framework for intervention design [2].

As part of a demonstration of our iterative framework (Fig 1), the parameters used in our study were specified using summary statistics and other data collected by the authors during previous studies as reported on in [1,2] on day laborer communities in the Houston and Austin Texas metropolitan areas. Those studies resulted in, among other results, the following parameters and empirical estimates involved in the decision process of the day-laborer communities germane to the RL formulation of this study; see Table 1.

It is important to acknowledge that the data collection in these previous, artifactual studies involved interviews with a relatively small ($n < 40$) non-random sample of immigrant workers. We recognize that this sample size and non-random selection limit the generalizability of these specific parameter values. However, for the purpose of demonstrating the reinforcement learning methodology and its potential for policy development, these empirically derived parameters serve as a foundational starting point, reflecting observed real-world dynamics. Future work will focus on performing a more formal sensitivity analysis to rigorously assess how uncertainty in these estimated parameters propagates through the model and influences policy outcomes. Furthermore, per our iterative framework (Fig 1), subsequent research phases will involve collecting additional, larger-scale empirical data to validate the model's predictions and refine the policies derived from the reinforcement learning approach.

Laborers, when they find work, typically earn between $80 and $150 per day and are assumed to experience costs of about $60 per day for bare necessities. They experience some degree of wage theft about 30% of the time with the degree of theft averaging about 25% of the wages agreed to at the beginning of the job. Currently, reporting options exist

Table 1. Empirically estimated system parameters.

Job pay range ($ per day)	80–150
Cost of living ($ per day)	60
Probability of theft (for single-agent)	0.3
Theft percent range	0%–25%
Probability of report success	0.01
Punitive damages–nominal value	500

Empirically estimated system parameters from [1] used in the agent-based simulation environment.

https://doi.org/10.1371/journal.pcsy.0000079.t001

but are highly unlikely to produce an outcome that restores wages to a victimized laborer (we use a 1% success rate). Suppose a report of wage theft produces a positive outcome for the laborer. In that case, there is also the chance that the employer will need to pay punitive damages to the laborer (we assume a $500 value for such damages).

3.2.2. Computational RL framework. The current study is focused on addressing complex decision-making problems in stochastic stationary environments using reinforcement learning as the primary policy development methodology. Specifically, the paper investigates the application of embedded discrete Markov chains to model decision processes in a real-world context, motivated by previous work by the authors.

Adopting a model-free approach, it is crucial to delineate the framework into two distinct components: the environment model (where data is collected from the real world or generated, as in the current study, from an agent-based simulation) and the reinforcement learning algorithm (where optimal policies are determined). The environment model encompasses the state space, action space, and the resultant rewards stemming from those actions. This study first analyzes the environment around a single laborer agent whose state evolves based on its actions. As discussed above, we then analyze the environment for the multi-agent scenario. Both scenarios involve a laborer receiving job offers with varying degrees of fairness, the possibility of theft, and potential reporting actions. The general process that is followed in this paper is expressed in Fig 6 below, which is elaborated on through the remainder of this section.

The reinforcement learning algorithm employs batch Q-learning to derive the optimal policy. What sets our reinforcement learning algorithm apart is its generality and adaptability. It can ingest data from diverse sources, whether from complex real-world sociological systems, synthetic data generated from complex, evidence-based, agent-based models (ABM) designed to mimic the real world, or simplified custom environments designed to generate essential data to train the algorithm and produce policies that can be further tested in the real world or sophisticated simulation environments. This adaptability allows the algorithm to dynamically adjust to changes in environmental input data, continually refining its decision-making capabilities. The methodology proposed in this study offers a flexible model approach and learning from various data sources. We can initiate the process with data from simplified environments and construct an initial model. As more real-world data becomes available, the model can be further refined and optimized, illustrating the iterative and adaptive nature of the framework in tackling decision-making complexities within sociological systems.

1. Custom Environment: The custom environment encapsulates the states, actions, and dynamics of the decision-making process. In the environment, when the laborer and employer are in the “idle” state:

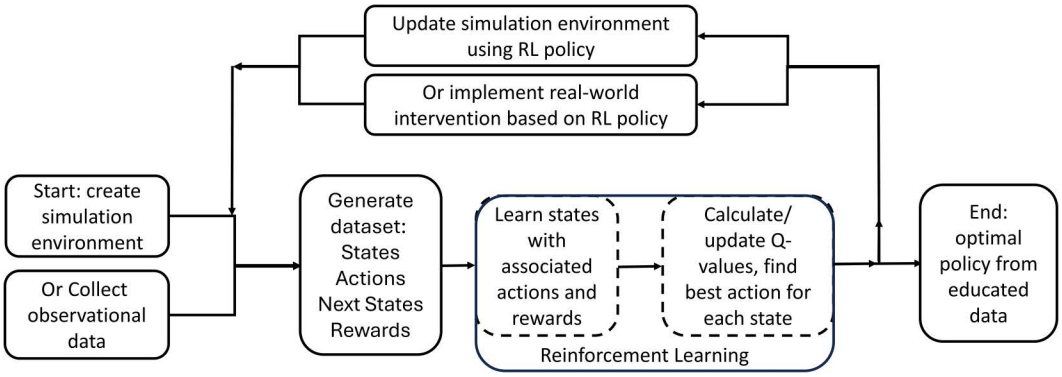

Fig 6. Extended Q-learning environment. A simulation environment generates states, actions, transitions, and rewards, which are learned by the reinforcement learning system. An optimal policy is found and used to direct the second round of the data generation process. The second round is then learned, and a new policy is established based on the previous one, indicating the optimal action at each state for a particular environment.

https://doi.org/10.1371/journal.pcsy.0000079.g006

- If the employer decides to offer a job and the laborer opts to “decline” the job offer, both the employer and the laborer face a penalty proportional to the cost of remaining idle, multiplied by the expected duration of remaining idle.
- Accepting a job offer involves assessing the probability of theft.
- If the employer decides to exploit, the laborer moves to the “workingTheft” state. In this case, the reward is computed as the daily pay minus a penalty related to the degree of theft, adjusted by the cost of living.
- Otherwise, if the employer is fair, the laborer transitions to the “workingFair” state, earning the daily pay minus the cost of living.

During employment in the “workingFair” or “workingTheft” states:

- Choosing “noAction” signifies continuing the current job without additional action. At the end of the employment period, the laborer returns to the “idle” state, with no immediate reward but the possibility of future job offers.
- In the case of working under theft conditions, opting to “report” the exploitation may or may not succeed, resulting in different consequences. If successful, the laborer receives compensation for the daily pay lost due to theft, in addition to punitive damages for the pain and suffering. Otherwise, the laborer experiences no immediate reward if the report fails, but the job still ends.

2. Simulation and Data Generation: The script conducts two phases of data collection. In the first pass, feasible actions for each state are determined by sampling experiences with random actions. Subsequently, the main data collection phase uses an epsilon-greedy strategy to balance exploration and exploitation for a specified number of simulations of the abovementioned decision process. The resulting experiences are collected as a data frame, which includes state-action pairs, rewards, and next states. This data is then processed by a separate function that learns the state space and sets of actions before going into the reinforcement learning brain to keep the RL algorithm model free. Algorithm 1 below shows how the code learns the state space that is then read into the reinforcement learning algorithm 2.

```
Algorithm 1: Learning the State Space and Action Sets
Data (data frame of state, action, reward, next state, where each
   row is a new observation, could be from an agent-based
   model, environment function, or real data)
Procedure Learn State Space (data)
  Initialize state_action_dictionary = {}
  For each row in the data frame
    state = row['State']
    action =    row['Action']
    If state is not in the state$_$action$_$dictionary:
      Initialize state_action_dictionary[state] = []
  Append state_action_dictionary[state] to include (action)
Return state_action_dictionary
```

3. Q-Learning Implementation: The Q-learning algorithm derives optimal policies in the defined environment. The Q-values are stored in a dictionary (Q) representing the expected cumulative rewards for state-action pairs. The Q-learning process involves iteratively updating the Q-values based on the experiences collected during the simulation. In our case study, the Q-values represent the net present value of the maximum expected income for the day laborer if he or she takes the action at that state and continues to make the best actions moving forward. Best, in this case, means maximizing expected income. The Q-values associated with the firm have very similar interpretations.

The Q-learning update equation for the single-agent framework uses the traditional format of Equation 1 in Section 4.4. The multi-agent Q-learning update equation is a slight variation where instead of the agent choosing the next action to take, the other player chooses the next action that is taken.

This Q-learning update equation for player i with opponent j is represented in equation 2 below.

$$Q_i(s,a) \;=\; Q_i(s,a) + \alpha(r_i \;+\; \gamma \;*\; Q_i(s', \underset{a'}{\text{argmax}}\, Q_j(s', a')) \;-\; Q_i(s,a)) \tag{2}$$

where the value agent *i* receives from the current state-action pair depends on the true belief of the action the opponent makes in the next round. The players then continue to take turns until the process is terminated.
The hyperparameters that are used in this study for the Q-Learning implementation include the initial learning rate (α) of 0.1 and the initial greedy action selection probability ($\in$) of 0.5, which both decay at a rate of 0.99 and both have minimum values of 0.01. We select out discount factor (γ) to be 0.9, a history window of 10 iterations. A sensitivity analysis measuring the effects of changing the initial values of α and $\in$ as well as δ on the system is conducted.

The reinforcement learning functions (reinforcement_learning for single-agent and reinforcement_learning_combine for the multi-agent) update the Q-values using the Q-learning update rule above based on the collected experiences. The parameters, alpha and gamma, control the learning rate and discount factor, respectively. The function supports the option to use a pre-existing Q-model, facilitating iterative learning. A key distinction of this algorithm is that the Reinforcement/Q learning processes have no knowledge of any underlying model, decision process, or game tree. Instead, this process's only inputs are the observed transition rewards after simulation or observing each state-action pair. The algorithm below outlines this process.

```
Algorithm 2: Reinforcement Learning
Data (learned state_action_dictionary, each observation lists the state,
  action, reward, and next state)
Procedure Reinforcement Learning (data, alpha, gamma, feasible actions, Q)
  If Q=NULL
    Initialize Q = {}
  For each experience observed in the data
    state=  experience["State"]
    action=  experience["Action"]
    reward=  experience["Reward"]
    next_sate=  experience["NextState"]
    NewQ[(state, action)] = currentQ(state, action) +
                            alpha * (reward+gamma * (max(Q(next_state, a)))
                            - currentQ(state, action)
                            where a is all feasible actions
Return NewQ
```

The Q-values obtained from the reinforcement learning process represent the learned policies for the decision-making process. In labor exploitation, the Q-values represent the net present value of expected income received when all future actions are chosen optimally. In the single-agent case, the same agent makes all the best actions. In the multi-agent setting, the competing agent determines the actions taken in the following state; then the process recursively alternates between the two agents. These Q-values provide insights into the optimal actions for each state, considering the stochastic nature of the environment. Higher Q-values suggest more favorable actions in corresponding states, indicating higher expected cumulative rewards. Lower or negative Q-values indicate less favorable actions for a state. A Q-value of zero typically implies that the action is neutral or does not contribute significantly to maximizing the expected cumulative reward.

By analyzing the Q-values, one can identify the optimal policy for the agent, which involves selecting actions that maximize the expected cumulative reward in each state. The RL algorithm updates these Q-values iteratively based on the observed rewards and transitions, aiming to converge toward an optimal policy over time. The results can be analyzed to understand the impact of fairness, theft, and reporting on the cumulative rewards of the laborer.

While batch Q-learning has well established convergence theoretically [32,50–52], in practice, tracking and establishing convergence is challenging and is the subject of ongoing research. Approaches generally focus on examining the observed learning rate of the algorithm, both visually and quantitatively, as well as hyperparameter tuning for α (for the learning rate) and $\in$ (for the likelihood of exploring a different policy than the current best action.)

5.2.3. Batch Q-learning convergence diagnostics

Following established approaches to tracking learning rates and convergence discussed in the literature, e.g., [32,50–52], we define three empirical criteria to monitor convergence of Q-values in our batch reinforcement learning setting, each using a fixed-size look-back window.

Mean Change. The average of the absolute differences between consecutive Q-values within the window. Convergence is indicated when this value falls below a threshold τ_{mean}.

$$\text{MeanChange}_t = \left(\frac{1}{W}\right) \sum_{k=t}^{t+W-1} \left|Q_k(s,a) - Q_{k-1}(s,a)\right| < \tau_{mean} \tag{3}$$

Variance of Q-Value Change. The variance of the absolute differences between consecutive Q-values within the window. Convergence is indicated when this value falls below a threshold τ_{var}.

$$\text{VarChange}_t = \text{Var}\left(\{\left|Q_k(s,a) - Q_{k-1}(s,a)\right|\}_{k=t}^{t+W-1}\right) < \tau_{\text{var}} \tag{4}$$

Percentage of Stable State-Action Pairs. The percentage of Q-value changes within the window that are below a small stability threshold τ_{stable}. Convergence is indicated when this percentage exceeds a high value (e.g., 95%).

$$\text{PercStable}_t = \frac{\sum_{k=t}^{t+W-1} I\left(\left|Q_k(s,a) - Q_{k-1}(s,a)\right| < \tau_{\text{stable}}\right)}{W} > 0.95 \tag{5}$$

Where I(·) is the indicator function.

Learning Rate and Exploration Rate Decay The learning rate α and exploration rate $\in$ decay over iterations (or runs) as follows:

$$\alpha \leftarrow \max(\alpha_{min},\ \alpha \cdot \text{decay\_rate})$$

$$\in \leftarrow \max(\in_{\text{min}},\ \in \cdot \text{decay\_rate})$$

Where α_{min} and $\in_{\text{min}}$ are the minimum allowed values for the respective rates, and decay_rate is the decay factor (e.g., 'epsilon_decay') which is set at 0.99.

Our results, presented next, are based on implementations of the mean absolute change and variance of the Q-value change for the sake of parsimony. In the current study, we decay α and $\in$ over the computational iterations such that the algorithm becomes less greedy and less likely to experiment with new policy candidates. We return to these considerations in our results and discussion sections.

4. Results

4.1. Single-agent formulation

In the single-agent environment, after running the program with the associated parameters with a sample size of 100, we collected Q values from the data creation process. The following results include an initial sample size of 100 as no

practical or significant effect that impacted the results was observed in increasing the initial sample step above 100 (1,000, 5,000, etc.). Then, we used those Q values as training data to go back into the environment, run again, and get the Q values after 1,000 iterations of batch Q-learning. To examine which parameters of our environment's structural model are influencing this near tie between the two Q values, we ran a sensitivity analysis. In Fig 7 below, we test the counterfactual effect of increasing the probability of success beyond the current likelihood of 0.001. The output represents the laborer's Q-values for reporting or not, expressed through the 1,000th iterations of batch-Q-learning performed at probabilities of 0.001, 0.01, and 0.02 for a successful report. The 95% confidence intervals provided were created using 1,000 bootstrap samples of 1,000 runs of the simulation.

We see above that as the probability of a successful report increases, so does the expected payoff to the day laborer, and thus, the Q-value of reporting increases approximately linearly. If the day laborer chooses not to report, the expected payoff also increases as the employer does not learn to stop stealing, and the laborer will be able to report successfully in the future, so the Q-value of No Action also increases. The overlapping of confidence intervals signifies a lack of statistical significance at those particular reporting success rates, indicating that the decision-maker does not have a clear optimal choice at most of 0.1% but does have a clear choice at 1% and 2%. As our algorithm chooses the highest Q-value at each state, it appears that holding all else fixed, a probability of reporting above 1 percent would remove this "indecision" for the decision-makers, and they would opt for reporting. This observation shows that the distance between the confidence intervals expands proportionally as the reporting success increases. This observation implies that systemic alterations influence behavioral change. It highlights the sensitivity of the policy to real-world systemic factors. Consequently, a non-substantial increase in reporting success is necessary to drive meaningful changes in decision-making and optimal policy formation in a simple environment where the laborer has no impact on the system.

4.2. Multi-agent formulation

We increased the realism in our analysis by including decision-making by both the laborer and the employer. In the multi-agent environment, after running the program with the empirically estimated parameters mentioned in the methods sections with 1,000 iterations of 100 samples, we collected Q-values from the data creation process. Then we used those Q-values as training data to go back into the environment, run again, and get the Q-values for both the employer and the day-laborer after batch Q-learning. Fig 8 below shows the Q-values of the employer's decision on whether or not to steal on the left and the employee's Q-values for their decision on whether or not to report a steal on the right. The top line in either graph indicates the optimal action to perform in that state. The width of the two lines represents a 95% confidence interval derived from 1000 bootstrap samples of 1000 runs of the simulation.

Holding all else the same, performing a similar analysis as the single agent case by increasing the probability of success results in the lines swapping places for the employer, and the distance between the Q-values of the employee's response increases. Fig 8 displays the Q-values for their decision whether or not to report a steal on the left and the employer's decision on whether or not to steal on the right for the empirically estimated 0.1% chance of reporting success, a 1% chance, and a 2% chance. The y-axis represents the Q-values, and the x-axis represents the number of iterations. Again, the overlapping of confidence intervals signifies a lack of statistical significance at those particular reporting success rates, indicating that only the laborer does not have a clear optimal choice at 0.1% but does at all other probabilities. It is important to note that a range of values exists where it is optimal for the laborer to report a steal while it's still optimal for the employer to steal (e.g., less than 2%). As the likelihood of reporting success increases (here as low as 2%), the policy converges to employers not stealing and laborers reporting wage theft that they experience.

4.3. Key findings

Following 1,000 runs of the simulation, each with 1,000 iterations, Table 2 below presents the average Q-values of interest after the 1,000 iterations for both the single-agent and multi-agent formulations. The table also includes the iteration at

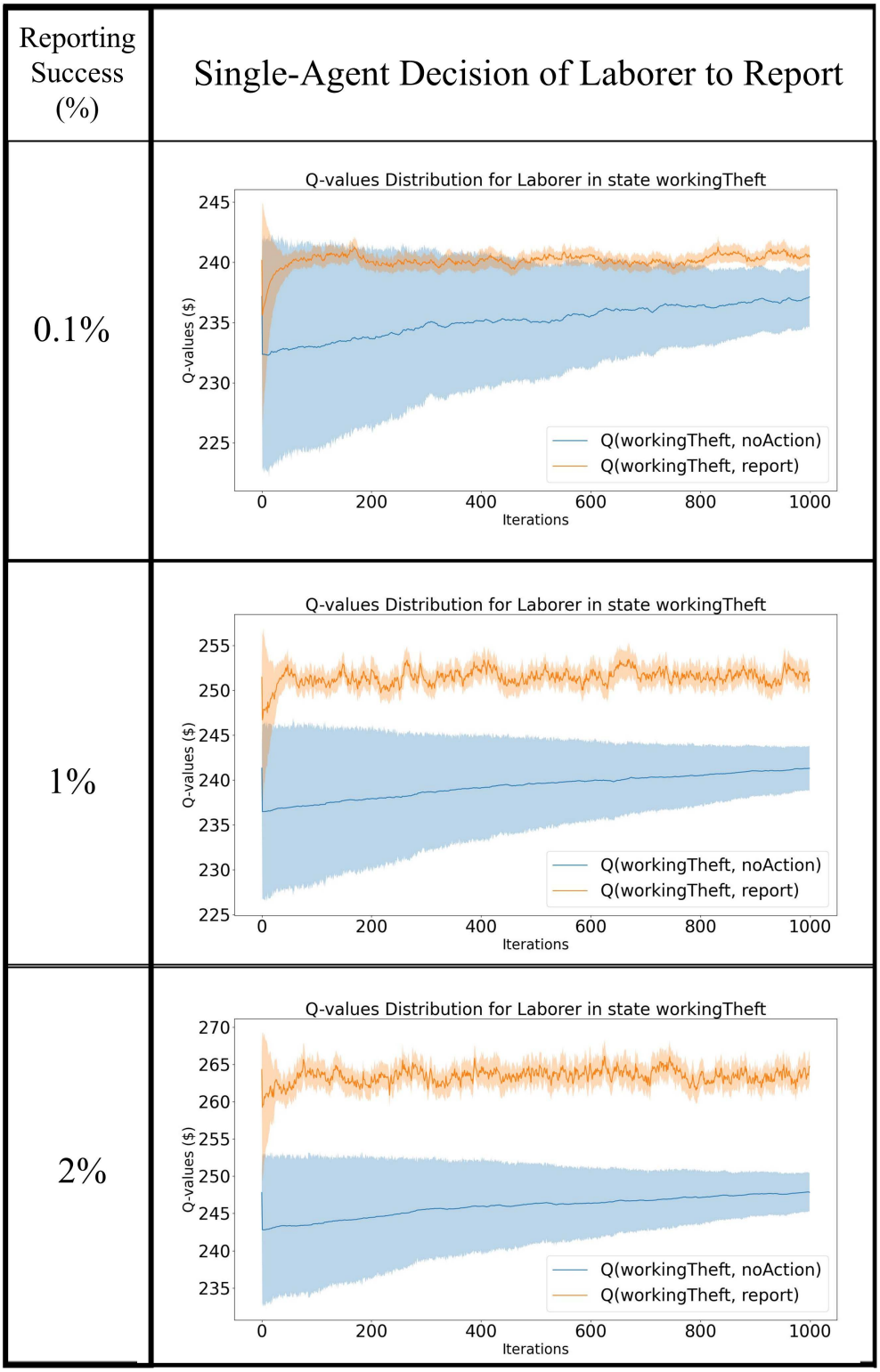

Fig 7. Single Agent Q-Values. The variations in Q-values corresponding to two actions, "Report" and "NoAction," within the "WorkingTheft" state across different probability values of report success. These results are generated by the Python code S1 File provided in the Supplement.

https://doi.org/10.1371/journal.pcsy.0000079.g007

Fig 8. Multi-Agent Q Values. Illustration of the employee's Q-values for their decision on whether or not to report a steal on the left and the employer's decision on whether or not to steal on the right for various values of the probability of reporting success.

https://doi.org/10.1371/journal.pcsy.0000079.g008

Table 2. Q-values and convergence statistics by agent action and reporting probability.

Type	Action	0.1%			1%			2%		
		Q-Value	Mean Conv.	Var Conv.	Q-Value	Mean Conv.	Var Conv.	Q-Value	Mean Conv.	Var Conv.
Single-Agent	Report	238.614	3014	5429	245.603	32134	44624	263.942	44624	32134
	No Action	238.159	11	11	243.536	45	45	248.437	43	43
Multi-Agent	Employer: Steal	104.663	818	5958	93.880	679	843	78.657	843	679
	Employer: No_Steal	99.220	2287	192	90.871	310	1906	84.973	342	224
	Laborer: Report	96.198	137	137	112.671	69	69	161.369	72	72
	Laborer: No_Report	92.438	828	78	101.398	725	102	132.271	616	45

https://doi.org/10.1371/journal.pcsy.0000079.t002

which the respective Q-value converged via mean absolute change, and variance of Q-value change, allowing for up to 10,000 iterations, both with a threshold of 0.01. The ordinality of the Q-values is the most important part of understanding the optimal action any agent should take in any scenario. In cases like these, where the Q-values have an interpretation of the net present value of wages, the cardinality can be used to calculate effect sizes. For example, in the multi-agent framework, an increase in the successful reporting rate from 0.1% to 1% results in a $15 increase to the worker for reporting on average and an $8 increase for not reporting. More significantly, the effect of raising the success rate from 1% to

2% results in an additional $49 increase to the worker for reporting on average. This larger increase is partly due to the employer's optimal action shifting to not stealing at 2%, as shown earlier in Fig 8. This is also reflected in Table 2, where the change from the successful reporting from 0.1% to 1% brings the q values from the employer closer together and the Q-values swap order at 2%. This small incremental change, requiring only a 2 percentage point increase in reporting success, can likely be achieved through a nudge, such as advocacy groups or public service announcements. This continues in the discussion.

The Python file that runs the simulations and provides the resulting graphics is provided as the Supplemental document S1 File, and written information of the Python file's contents can be found in the following GitHub repository: https://github.com/EvanAldrich/RL-ProductiveEmployment. Sensitivity analyses were also conducted to examine the effects of the learning parameters' values (alpha, gamma, epsilon) and environmental parameters (pTheft, propDegreeTheft) on both the single-agent and multi-agent systems, which are presented in the supplemental document S1 Appendix.

5. Generalizing the model

The problem we address in this study, wage theft experienced by day laborers, has been presented in a simplified fashion but has allowed us to examine dynamics that illustrate that the system is complex and uncertain. In this section, we use those dynamics to develop a more generalized framework that can represent such systems more completely. Specifically, the day laborer system presented in this study is an example of a stochastic discrete event dynamic system (SDEDS). A useful type of model for describing SDEDS that we further develop here as a continuation of this work is the class of generalized semi-Markov schemes, which include an embedded stochastic process referred to as a GSMDP [53,54].

In the current schema, a generalized semi-Markov scheme (GSMS) is defined by a quintuple (S,E,A,p,R) where:

- S denotes a countable set of system states;
- E denotes a countable set of events;
- $A(s)$ denotes the subset of E whose members are active in the current state, $s \in S$;
- $p(\cdot\,|s, e)$ is a conditional probability measure over the states in S with the interpretation that $p(s'|s,e)$ is the conditional probability that a transition is made from state s to state s' due to e; $s,s' \in S$, $e \in E$;
- R denotes the set of rewards for state s and event e.

For event $e \in E$ in state $s \in S$, we denote $L = \lambda_e(s)$ as a collection of nonnegative speeds that characterize the rate at which e consumes its lifetime in state s.

This addition of events could allow for a more nuanced understanding of decision-making where agents respond to events in the state space rather than simply reacting to the system existing in a particular state. In each macrostate, many events are taking place, and these events live through their lifetime as the agents interact in the system. The simplest dynamic involves one event reaching the end of its lifetime first, denoted the trigger event, that causes the system to change macrostate. A policy is then designed to act in response to that event. The system can thus evolve naturally. However, a policy either reacts to that trigger, moving the agents toward a particular reward followed by a transition to the next state where some of the events continue their life, and the trigger either does not exist anymore or is reset. In this schema, it is also possible for an event to be triggered by the policy proactively to move the system in a more desirable direction by preventing a less desirable natural trigger. The GSMP has been formulated previously but has received little attention recently [53–57]. Such a framework seems well suited for future efforts for systems with the complexity we are addressing.

For the current study, even in our simplified case study, during a job, the laborer may realize that the prospects of being paid fairly are low based on interactions with the employer during work performance. As we have observed in

our empirical research, the laborer can decide to leave the job proactively at that time, thus triggering the state change. Although the decision in this example introduces great uncertainty about whether they will be paid at all, laborers make this choice to preserve their sense of justice. While this is a complicated scenario that we have not modeled in the present study involving multicriteria decision-making, we plan to address such proactive, preventative decisions in future research. For our purposes here, this is an example of when the end of the job is a consequence of a decision by one of the agents in the system, as opposed to the job ending (state changing) naturally. The system generally moves to a new macrostate, characterized by all active events evolving in that state. In our case study, the laborer is once again idle and looking for their next job. In our model, for such instances, the policy activates a trigger, prompting the agent to act in both reactive and proactive manners. In general, mathematically, events compete based on their lifetime according to the rates, L, and the associated joint probability of triggering, potentially carrying over residual life from the previous state. See Fig 9 for details of this reactive/proactive event/action dynamic.

Interestingly, if we look back to our case study example in Fig 2, we see that events are inherently taking place to trigger state changes with different events serving as triggers. The discrete Markov chain formulation aggregates these events into uncertain state transitions and those stochastic state changes that need to be learned from the data used to train the RL algorithm. Specifically, at the start of our process, the day-laborer remained idle until a position was offered to them; this is an event triggered by the state change observed as a transition from idle to the unobserved state "position offered." Additionally, once the laborer is working, the next state change is either the job ending or the employer stealing wages. Our decision of interest in the case study of the choice to report relies on the event of theft occurring early enough for the state change to take effect. At the end of the decision process, outside of the decision maker's control, is the result of litigation of the report and whether punitive damages or the initial pay is given to the day laborer. Although these two outcomes do not have different next-state triggers, they impact the resulting reward in the decision process.
The Q-learning update equation might now become

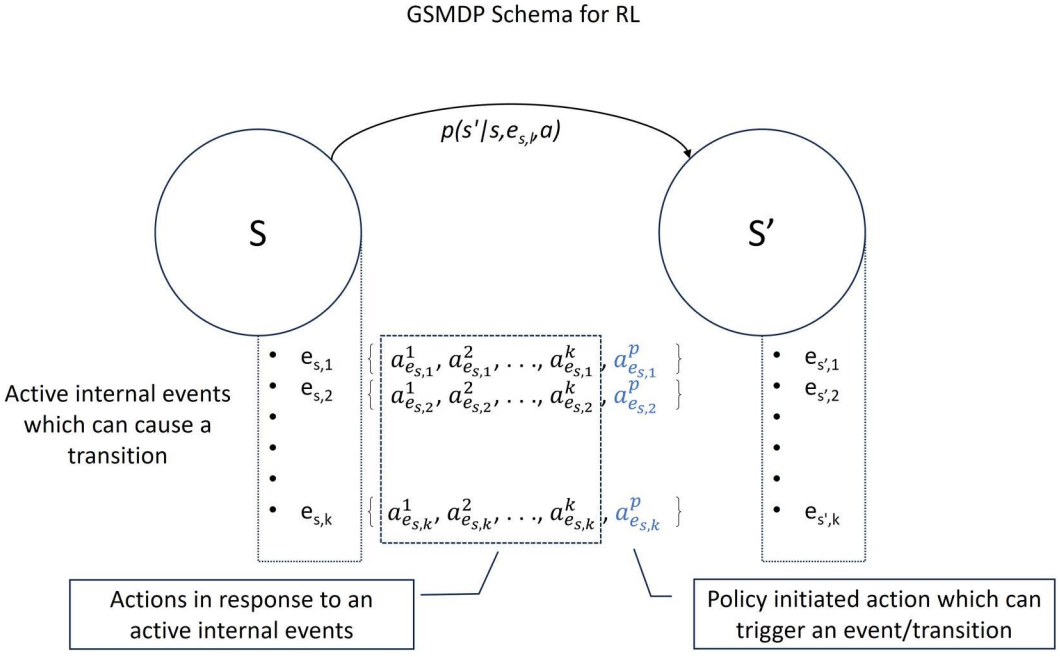

Fig 9. Event Supplementation of Macro States. Illustration of how macro states can be supplemented with events associated with microstate processes and associated reactive or proactive actions.

https://doi.org/10.1371/journal.pcsy.0000079.g009

$$Q(s,e,a) \;=\; Q(s,e,a) + \;\alpha(r_e(s) + \;\gamma \;*\; max(Q(s',e',a')) \;-\; Q(s,e,a)) \tag{6}$$

The hyperparameters retain the same meaning as before: α ($0 < \alpha \leq 1$) denotes the learning rate and γ represents the discount factor. The value of $max(Q(s',e',a'))$ now reflects the maximum Q-value for the next state-event-action tuple. In this setting, $r_e(s)$ signifies the immediate reward associated with the state s and the event e.

A more formal implementation of batch Q-learning with this extended schema would involve a indexing through a nested data dictionary that contains the observed tuples of training data. The extended schema expands the Markov chain's tuple to include events associated with actions taken that result in the next state. The particulars of each event can result in the same action having different outcomes, as we see with reporting in the case study presented. This inclusion requires a modification of the current schema to account for conditional probability measures over the reward space due to events and current states. As these events play a major role in the realistic nature of the decision process, a deeper look into events and the time between events is warranted.

Thus, optimal policies for GSMPs involve managing both the transition probabilities and the rate at which events utilize their lifetimes. For example, the decision to report or not may change depending on how long it has been since the work ended. From a control perspective, it is useful to view events as either corresponding to the initiation or termination of an underlying process in the GSMDP. Policies now control events that create a process lifetime or terminate a process lifetime (initiated control and initiated events), in addition to reacting to trigger events arising from the active event set (natural control and natural events). Input mechanisms for control actions influence the control state transition probabilities or the rate at which the system generates events. The GSMDP is expanded to include: 1) a control space which is the union of natural, initiated, and rate control actions, and 2) a general input mechanism which includes both reactive and preemptive actions.

As with the MDP, the GSMDP is formed from the GSMS by augmenting it with an action space U, forming (S,E,A,p,R,U). The control space, U, now has a class of actions that initiate events (denoted U^i) and a class of actions that allow the system to continue generating an event from the active event set (denoted U^n), with $U = U^n \cup U^i$. Similarly, A(s) can be partitioned into $A^n(s)$ and $A^i(s)$. Control can now be a function of the trigger event in addition to the supplemented state, i.e., $u = u(s,a,e^*)$. The transition probabilities, $p(s'|s,e^*,u)$, are interpreted as the conditional probability that a transition is made from state s to state s' due to event e^* and control action u. Similarly, $F_e(y|s,a,u)$ is the conditional residual lifetime distribution of event e conditioned on control action u and the age of event e, and $F_e(y|s,u)$ is the equilibrium residual lifetime distribution for event e conditioned only on control action u.

The consumption of a lifetime for event e is scaled by the rate $\lambda_e(s)$, i.e., $t * e_{\lambda e(s)}t$. There can be at most countably many different scaling rates $\lambda_e(s)$. This implies that controlling the scaling rate $\lambda_e(s)$ for an event e corresponds to initiating a transition to a state that differs from the current state only by the rate at which that event evolves. Thus, rate control is incorporated by expanding the model specification for the system to include the appropriate rate states and the appropriate initiation actions in the "initiate" control class. This approach leads to further expansion of the state space. Rate scaling can also be directly incorporated into the control policy via $F_e(y \mid s,a,u)$, where now u is a function of s, a, and $\lambda_e(s)$.

The discussion above concerning the GSMP falls within the modeling and environmental context of our generalized adaptive research framework. The richer the model and the more it can account for real-world events, time, and accurate decision-making, the more instructive the results may be. The data provided by a simulated environment would then be read into a reinforcement learning brain, which iterates toward the optimal strategy for a decision maker given researcher-defined assumptions of their maximization problem. In our presented case study, the utility being maximized was a Q-value representing the present value of maximal expected payoffs in dollars if the decision maker continues to make the best strategies moving forward. This optimization process is used to determine the optimal policy which provides the decision maker's best action to take at each state. The learned optimal policy would then be used to update the simulated environment or in the real world to implement actual policies to promote a desirable sociological outcome.

6. Discussion

This study used batch Q-learning to examine the decision process of laborers and employers as they engage in fair and unfair work toward the goal of designing sustainable full employment and a fair work environment. We have used an adaptive learning framework (see Fig 1) to design increasingly informed policies to manage the interactions between laborers and employers. In an initial single-agent formulation, the methods have replicated the real-world situation so common in precarious employment where the best option for employers is to offer the job and for laborers is to accept jobs offered to them. However, with the current inadequate reporting system, employers receive a higher payoff when they decide to steal wages. This is also reflected in the inconsistent results seen in the decision path of the laborer to report the wage theft or not. This may be due to the current low rates of a report being successful in accurately reflecting the real-world observations of cases not being brought up or the lack of awareness regarding the reporting process [2]. The massive repercussions of reporting unsuccessfully and the employer finding out could result in long periods without income and may be too costly to consider.

In a second, multi-agent formulation, the methods have demonstrated the efficacy of interventions that let the laborers know how and where to file a report using advocates and public service announcements to increase the likelihood of a successful report [2]. The consequences of such interventions are seen to decrease an employer's willingness to steal, either probabilistically or as a decision by an agent, increasing the well-being of the day laborers. Following the process of Fig 1, the authors are continuing to collect information through additional interviews on trust between laborers and employers to generate a more accurate representation of the system compared to the current version based on data from the artifactual experiments used thus far.

More specifically, a unique ability of a model-free Q-learning algorithm such as the one developed in this study is that the Q-learning updates itself to any change in the environment. Therefore, additional states and actions could be added, and the Q-learning model would read in the data frame from the modified environment and show the Q-values associated with the action/state pair without directly needing to know the data frame beforehand. The corresponding Q-value can be calculated as long as the associated data frame provides data starting with the current state, event, action, and next state. This flexibility of Q-learning allows for many applications similar to the day-laborer community, where it reads in either real data or a simulated environment. Additions could be made to our data creation process to develop a more robust understanding of the inconsistencies. And as appropriate, more complex and scaled-up data collection protocols could provide similar increases in our results' robustness. While our computational model has simplified the real-world scenario, it highlights numerous complexities that demand resolution. A more intricate model is warranted to address these challenges effectively as part of the proposed iterative intervention design framework.

In the computational approach used in this paper, which simulates a laborer-employer interaction, the reward values are based solely on monetary returns. However, even in this simplified model, multiple objectives govern the system, and these objectives may often conflict with each other. For instance, while accepting a job offer may result in immediate net financial gain (possibly reduced by theft), decision-making in such an environment may also impact job satisfaction, work-life balance, and overall well-being.

Recognizing the multi-faceted nature of decision-making processes, there is a growing interest in expanding RL utility functions to incorporate multiple objectives. This integration of multiple objectives into the utility function enables agents to make more informed and balanced decisions, ultimately leading to better outcomes. This approach aligns with the broader research direction in multi-objective reinforcement learning (MORL) frameworks, which aim to address the challenges posed by decision-making in complex, multi-dimensional environments [58].

Similarly, in the context of the Markov decision process, as explained in the theoretical framework section of the study, we can expand the decision framework to the more generalized semi-Markov decision process (GSMP) that incorporates subprocesses internal to the state that generate events, either naturally occurring or as part of a control policy [53,54,59]. The GSMP includes countable macro states, as examined here thus far, that is conjoined with a vector of such

subprocesses. These subprocesses are incorporated in the GSMP schema by tracking the time in between events in the same state, one of which will trigger an event that causes a macro-state transition. We included a simple implementation of this dynamic in our current model by introducing an expected amount of time idling before a job is offered as a function of the probability of being offered work.

The notion of macro-states as a representation of a complex system with detailed underlying event-based dynamics has broad application. As an example that extends the current study on labor exploitation, the authors are involved with examining sustainable community development in a setting where human/wildlife conflict is common. In this setting, a community is comprised of many agent types, including those with a governing role, members who live and work cooperatively in the community, as well as members whose behaviors are at odds with broader community norms and goals [60]. As part of an ongoing study situated in remote south-eastern Africa in a region that spans national borders, society is tribally structured, notions of work and employment are even more informal than between day laborers and the employers who hire them as manifest in our current case study, and resources are scarce [61].

Such scarcity is a major driver of illicit activities, principally in poaching and illegal wildlife trade. Tribal leaders must manage both community and wildlife prosperity. Communication within and between communities is hierarchical. In this setting, the state space could represent levels of community health or well-being. Actions cover licit community activities like farming, herding, and wildlife management, as well as illicit activities like wildlife poaching and illegal wildlife trade. Decisions would span degrees of community investment of human capital on licit activities and the avoidance of illicit activities. Like our study of wage theft, reporting illicit activities to governing agencies is inefficient and has a low likelihood of penalties for perpetrators. Events would include naturally occurring environmental dynamics as well as various inter-agent network dynamics that produce economic and well-being outcomes as well as those that respond to or prevent illicit activity. Desired policies would help tribal leaders allocate scarce resources and forms of capital toward the goal of broader community prosperity and well-being while minimizing the need for community members to engage in illicit activities.

Supporting information

S1 File. Python script.
(PDF)

S1 Appendix. Sensitivity analysis.
(PDF)

Acknowledgments

The authors acknowledge the contributions of Nayan Vashisit, MSE, in the programming for the single-agent version of the RL framework and feedback on an earlier version of the paper. We also acknowledge the influence of the IC^2 Institute's (https://ic2.utexas.edu/) strategic focus on innovating wellbeing and responsibly developing artificial intelligence and decision support systems for deployment in health care settings.

Author contributions

Conceptualization: Matt Kammer-Kerwick.

Data curation: Matt Kammer-Kerwick, Evan Aldrich.

Formal analysis: Matt Kammer-Kerwick, Evan Aldrich.

Funding acquisition: Matt Kammer-Kerwick.

Investigation: Matt Kammer-Kerwick, Evan Aldrich.

Methodology: Matt Kammer-Kerwick, Evan Aldrich.

Project administration: Matt Kammer-Kerwick.

Resources: Matt Kammer-Kerwick.

Software: Matt Kammer-Kerwick, Evan Aldrich.

Supervision: Matt Kammer-Kerwick.

Validation: Matt Kammer-Kerwick, Evan Aldrich.

Visualization: Matt Kammer-Kerwick.

Writing – original draft: Matt Kammer-Kerwick, Evan Aldrich.

Writing – review & editing: Matt Kammer-Kerwick, Evan Aldrich.

References

1. Takasaki K, et al. Wage theft and work safety: immigrant day labor jobs and the potential for worker rights training at worker centers. Journal of Labor and Society. 2022;25(2):237–76. https://doi.org/10.1163/24714607-bja10066
2. Kammer-Kerwick M, Yundt-Pacheco M, Vashisht N, Takasaki K, Busch-Armendariz N. A Framework to Develop Interventions to Address Labor Exploitation and Trafficking: Integration of Behavioral and Decision Science within a Case Study of Day Laborers. Societies. 2023;13(4):96. https://doi.org/10.3390/soc13040096
3. Shteingart H, Loewenstein Y. Reinforcement learning and human behavior. Curr Opin Neurobiol. 2014;25:93–8. https://doi.org/10.1016/j.conb.2013.12.004 PMID: 24709606
4. Battista W, Romero-Canyas R, Smith SL, Fraire J, Effron M, Larson-Konar D, et al. Behavior Change Interventions to Reduce Illegal Fishing. Front Mar Sci. 2018;5. https://doi.org/10.3389/fmars.2018.00403
5. Frazzoli C. The vulnerable and the susceptible: The weight of evidenza to stop exploiting activities generating toxic exposures in unprotected and deprived countries. J Glob Health. 2021;11:03046. https://doi.org/10.7189/jogh.11.03046 PMID: 33828831
6. Weintraub WS, Daniels SR, Burke LE, Franklin BA, Goff DC Jr, Hayman LL, et al. Value of primordial and primary prevention for cardiovascular disease: a policy statement from the American Heart Association. Circulation. 2011;124(8):967–90. https://doi.org/10.1161/CIR.0b013e3182285a81 PMID: 21788592
7. Fioramonti L, Coscieme L, Mortensen LF. From gross domestic product to wellbeing: How alternative indicators can help connect the new economy with the Sustainable Development Goals. The Anthropocene Review. 2019;6(3):207–22. https://doi.org/10.1177/2053019619869947
8. Macchia L. Governments should measure pain when assessing societal wellbeing. Nat Hum Behav. 2023;7(3):303–5. https://doi.org/10.1038/s41562-023-01539-3 PMID: 36765169
9. Coscieme L, Sutton P, Mortensen LF, Kubiszewski I, Costanza R, Trebeck K, et al. Overcoming the Myths of Mainstream Economics to Enable a New Wellbeing Economy. Sustainability. 2019;11(16):4374. https://doi.org/10.3390/su11164374
10. Boufkhed S, Thorogood N, Ariti C, Durand MA. "They treat us like machines": migrant workers' conceptual framework of labour exploitation for health research and policy. BMJ Glob Health. 2024;9(2):e013521. https://doi.org/10.1136/bmjgh-2023-013521 PMID: 38316464
11. Boufkhed S, Thorogood N, Ariti C, Durand MA. Building a better understanding of labour exploitation's impact on migrant health: An operational framework. PLoS One. 2022;17(8):e0271890. https://doi.org/10.1371/journal.pone.0271890 PMID: 35913945
12. Chigbu BI, Nekhwevha F. Exploring the concepts of decent work through the lens of SDG 8: addressing challenges and inadequacies. Front Sociol. 2023;8:1266141. https://doi.org/10.3389/fsoc.2023.1266141 PMID: 38053676
13. Capocchi L, Santucci J-F. Discrete Event Modeling and Simulation for Reinforcement Learning System Design. Information. 2022;13(3):121. https://doi.org/10.3390/info13030121
14. Bowling M, Veloso M. An analysis of stochastic game theory for multiagent reinforcement learning. Pennsylvania: School of Computer Science, Carnegie Mellon University. 2000.
15. Mosoetsa S, Stillerman J, Tilly C. Precarious Labor, South and North: An Introduction. Inter Labor Working-Class Hist. 2016;89:5–19. https://doi.org/10.1017/s0147547916000028
16. Tilly C. Worker Centers: Organizing Communities at the Edge of the Dream. Janice Fine. Administrative Science Quarterly. 2006;51(4):667–8. https://doi.org/10.2189/asqu.51.4.667
17. Standing G. The Precariat. Contexts. 2014;13(4):10–2. https://doi.org/10.1177/1536504214558209
18. Cranford CJ, Vosko LF, Zukewich N. Precarious employment in the Canadian labour market: A statistical portrait. Just Labour. 2003. https://doi.org/10.25071/1705-1436.164

19. Kalleberg AL. Precarious Work, Insecure Workers: Employment Relations in Transition. Am Sociol Rev. 2009;74(1):1–22. https://doi.org/10.1177/000312240907400101
20. Kalleberg L, Hewison K. Precarious Work and the Challenge for Asia. Title of the book or collection. Place of Publication: Publisher Name. 2023. American behavioral scientist. American Behavioral Scientist. 2013;57(3):271-288.
21. Dahan Y, Lerner H, Milman-Sivan F. Global Justice, Labor Standards and Responsibility. Theoretical Inquiries in Law. 2011;12(2). https://doi.org/10.2202/1565-3404.1275
22. Bright D, Koskinen J, Malm A. Illicit Network Dynamics: The Formation and Evolution of a Drug Trafficking Network. J Quant Criminol. 2018;35(2):237–58. https://doi.org/10.1007/s10940-018-9379-8
23. Konrad RA, Saeed K, Kammer-Kerwick M, Busaranuvong P, Khumwang W. "Fish-y" banks: Using system dynamics to evaluate policy interventions for reducing labor exploitation in the seafood industry. Socio-Economic Planning Sciences. 2023;90:101731. https://doi.org/10.1016/j.seps.2023.101731
24. Kellison B, et al. To the public, nothing was wrong with me: Life experiences of minors and youth in Texas at risk for commercial sexual exploitation. 2019. https://repositories.lib.utexas.edu/handle/2152/76119. 2019. 2023 January 4.
25. Iglesias-Ríos L, Marin DE, Moberg K, Handal AJ. The Michigan Farmworker Project: Development and Implementation of a Collaborative Community-Based Research Project Assessing Precarious Employment and Labor Exploitation. J Community Engagem Scholarsh. 2022;15(1):10.54656/jces.v15i1.466. https://doi.org/10.54656/jces.v15i1.466 PMID: 40821696
26. Iglesias-Rios L, Valentín-Cortés M, Fleming PJ, O'Neill MS, Handal AJ. The Michigan Farmworker Project: A Community-Based Participatory Approach to Research on Precarious Employment and Labor Exploitation of Farmworkers. Labor Stud J. 2023;48(4):336–62. https://doi.org/10.1177/0160449x231196227 PMID: 38939876
27. Caulkins JP, et al. A call to the engineering community to address human trafficking. National Academy of Engineering. 2019. https://www.nae.edu/216482/Fall-Issue-of-The-Bridge-on-Cybersecurity. 2024 November 25.
28. Galvin DJ. Deterring Wage Theft: Alt-Labor, State Politics, and the Policy Determinants of Minimum Wage Compliance. Perspect polit. 2016;14(2):324–50. https://doi.org/10.1017/s1537592716000050
29. Fair Labor Standards Act of 1938: Maximum Struggle for a Minimum Wage. https://www.dol.gov/general/aboutdol/history/flsa1938. 2024 November 9.
30. Fine JR. Worker Centers: Organizing Communities at the Edge of the Dream. Cornell University Press. 2006.
31. Theodore N. Day-Labor Worker Centers: Advancing New Models of Equity and Inclusion in the Informal Economy. Economic Development Quarterly. 2023;37(4):363–74. https://doi.org/10.1177/08912424231165004
32. Sutton RS, Barto AG. Reinforcement Learning, Second Edition: An Introduction. MIT Press. 2018.
33. Park YJ, Cho YS, Kim SB. Multi-agent reinforcement learning with approximate model learning for competitive games. PLoS One. 2019;14(9):e0222215. https://doi.org/10.1371/journal.pone.0222215 PMID: 31509568
34. Osborne J., Rubinstein A. A course in game theory. 12 ed. Cambridge, Mass.: MIT Press. 1994.
35. Abdoos M. A Cooperative Multiagent System for Traffic Signal Control Using Game Theory and Reinforcement Learning. IEEE Intell Transport Syst Mag. 2021;13(4):6–16. https://doi.org/10.1109/mits.2020.2990189
36. Lipowska D, Lipowski A. Emergence of linguistic conventions in multi-agent reinforcement learning. PLoS One. 2018;13(11):e0208095. https://doi.org/10.1371/journal.pone.0208095 PMID: 30496267
37. Huang Z, Tanaka F. MSPM: A modularized and scalable multi-agent reinforcement learning-based system for financial portfolio management. PLoS One. 2022;17(2):e0263689. https://doi.org/10.1371/journal.pone.0263689 PMID: 35180235
38. Tampuu A, Matiisen T, Kodelja D, Kuzovkin I, Korjus K, Aru J, et al. Multiagent cooperation and competition with deep reinforcement learning. PLoS One. 2017;12(4):e0172395. https://doi.org/10.1371/journal.pone.0172395 PMID: 28380078
39. Kim JI, Lee YJ, Heo J, Park J, Kim J, Lim SR, et al. Sample-efficient multi-agent reinforcement learning with masked reconstruction. PLoS One. 2023;18(9):e0291545. https://doi.org/10.1371/journal.pone.0291545 PMID: 37708154
40. Shapley LS. Stochastic Games. Proceedings of the National Academy of Sciences. 1953;39(10):1095–100. https://doi.org/10.1073/pnas.39.10.1095 PMID: 16589380
41. Solan E, Vieille N. Stochastic games. Proc Natl Acad Sci U S A. 2015;112(45):13743–6. https://doi.org/10.1073/pnas.1513508112 PMID: 26556883
42. Doraszelski U, Escobar JF. A theory of regular Markov perfect equilibria in dynamic stochastic games: Genericity, stability, and purification. Theoretical Economics. 2010;5(3):369–402. https://doi.org/10.3982/te632
43. Moutinho V, Silva A. Does technological progress, capital, labour, and categorical economic policy uncertainty influence unemployment? Evidence from the USA. Applied Economics. 2025;57(3):301–16. https://doi.org/10.1080/00036846.2024.2303618
44. Ghosh D, Sharma A, Shukla KK, Kumar A, Manchanda K. Globalized robust Markov perfect equilibrium for discounted stochastic games and its application on intrusion detection in wireless sensor networks: Part I—theory. Japan J Indust Appl Math. 2019;37(1):283–308. https://doi.org/10.1007/s13160-019-00397-9

45. Clempner JB, Poznyak AS. Modeling the multi-traffic signal-control synchronization: A Markov chains game theory approach. Engineering Applications of Artificial Intelligence. 2015;43:147–56. https://doi.org/10.1016/j.engappai.2015.04.009
46. Archibald TW, Possani E. Investment and operational decisions for start-up companies: a game theory and Markov decision process approach. Ann Oper Res. 2019;299(1–2):317–30. https://doi.org/10.1007/s10479-019-03426-5
47. Peters H. Game Theory: A Multi-Leveled Approach. Berlin, Heidelberg: Springer. 2015. https://doi.org/10.1007/978-3-662-46950-7
48. Watkins CJCH, Dayan P. Q-learning. Mach Learn. 1992;8(3–4):279–92. https://doi.org/10.1007/bf00992698
49. Lange S, Gabel T, Riedmiller M. Batch Reinforcement Learning. Adaptation, Learning, and Optimization. Springer Berlin Heidelberg. 2012. 45–73. https://doi.org/10.1007/978-3-642-27645-3_2
50. Hu J, Yang X, Hu J-Q, Peng Y. A Q-learning algorithm for Markov decision processes with continuous state spaces. Systems & Control Letters. 2024;187:105782. https://doi.org/10.1016/j.sysconle.2024.105782
51. Fu J, et al. Diagnosing bottlenecks in deep Q-learning algorithms. In: Proceedings of the 36th International Conference on Machine Learning, 2019. 2021–30. https://proceedings.mlr.press/v97/fu19a.html
52. Xie T, Jiang N. Q* Approximation Schemes for Batch Reinforcement Learning. 2023.A theoretical comparison. In: Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI). PMLR. In: Proceedings of Machine Learning Research, 2020. 550–9. https://proceedings.mlr.press/v124/xie20a.html
53. Kammer-Kerwick M. Controlled generalized semi-Markov processes: A framework for nearoptimal control of stochastic discrete event dynamic systems. The University of Texas at Austin. 1993.
54. Younes HLS, Simmons RG. Solving generalized semi-markov decision processes using continuous phase-type distributions. In: Proceedings of the 19th National Conference on Artificial Intelligence (AAAI), 2004. 742–7.
55. Zhuang W, Li MZF. Monotone optimal control for a class of Markov decision processes. European Journal of Operational Research. 2012;217(2):342–50. https://doi.org/10.1016/j.ejor.2011.09.021
56. Ghosh MK, Saha S. Optimal Control of Markov Processes with Age-Dependent Transition Rates. Appl Math Optim. 2012;66(2):257–71. https://doi.org/10.1007/s00245-012-9171-3
57. Kammer-Kerwick M, Busch-Armendariz N, Talley M. Disrupting illicit supply networks: New applications of operations research and data analytics to end modern slavery. 2018. https://doi.org/10.15781/7983-6q55
58. Hayes CF, et al. A practical guide to multi-objective reinforcement learning and planning. Autonomous Agents and Multi-Agent Systems. 2022;36(1):26.
59. Yu H, Mahmood AR, Sutton RS. On Generalized Bellman Equations and Temporal-Difference Learning. Lecture Notes in Computer Science. Springer International Publishing. 2017. 3–14. https://doi.org/10.1007/978-3-319-57351-9_1
60. Kammer-Kerwick M, Takasaki K, Kellison JB, Sternberg J. Asset-Based, Sustainable Local Economic Development: Using Community Participation to Improve Quality of Life Across Rural, Small-Town, and Urban Communities. Appl Res Qual Life. 2022;17(5):3023–47. https://doi.org/10.1007/s11482-022-10051-1 PMID: 35756429
61. Aguirre AA, Gore ML, Kammer-Kerwick M, Curtin KM, Heyns A, Preiser W, et al. Opportunities for Transdisciplinary Science to Mitigate Biosecurity Risks From the Intersectionality of Illegal Wildlife Trade With Emerging Zoonotic Pathogens. Front Ecol Evol. 2021;9. https://doi.org/10.3389/fevo.2021.604929