Optimizing Workforce Allocation through Reinforcement Learning: an automatic approach to Human Resource Management

Valentina Tonazzo, Marina Ceccon, Francesco Boscolo Meneguolo and Gian Antonio Susto*

Abstract— **Workforce allocation is a key challenge in modern companies, where dynamic project demands and diverse employee profiles must be managed effectively. We model this problem as a sequential decision-making task and apply Reinforcement Learning (RL) within a realistic simulation environment. Employees are characterized by skill sets, expertise, and availability, while projects have specific skill and availability requirements. Using Proximal Policy Optimization (PPO), an RL agent learns adaptive assignment strategies under varying conditions, including workforce shortages and surpluses. A fairness-aware reward function is employed, which discourages repetitive or unbalanced assignments. Experimental results demonstrate both allocation efficiency and fairness, underscoring the potential of reinforcement learning for responsible, data-driven workforce management.**

I. INTRODUCTION

Efficient workforce allocation is a critical challenge for modern organizations [10], especially in highly dynamic sectors such as consulting, where projects demand specific combinations of skills, expertise, and timing [13]. As a core element of strategic human capital management, Strategic Workforce Planning (SWP) aims to align an organization's talent resources with its long-term objectives, ensuring the right people are in the right roles at the right time [5].

However, traditional SWP methods [9], [15], [14], often based on static models or rule-based heuristics, struggle to capture the complexity of real-world human resource management, in the face of fluctuating availability and multi-dimensional constraints. These methods typically operate at an aggregate level and are ill-suited to handle fine-grained, real-time decision-making, particularly when individual employee attributes and time-sensitive project demands must be coordinated dynamically [1].

In this work, we introduce a realistic, custom-built simulation environment designed to reflect the operational characteristics of a consulting company. Moreover, we propose a Reinforcement Learning (RL) approach to model and solve the workforce allocation problem, leveraging the adaptability of intelligent agents to dynamically optimize assignments. Employees are modeled with binary skill vectors, expertise levels, costs, and time-based availability; projects are encoded with skill requirements, budget constraints, and activity timelines. Crucially, our formulation captures the temporal interdependencies inherent in workforce allocation: once a worker is assigned to a task, they become unavailable for a period of time, introducing a need for anticipatory planning. This shifts the problem from a static assignment task to a sequential decision-making process, where future task requirements and workforce availability must be taken into account.

A key innovation of this work lies in the design of the reward function, which incorporates not only efficiency-related metrics but also explicit fairness constraints, penalizing redundant assignments, and promoting equitable resource usage across the employee pool. This dimension is often overlooked in the related literature, where performance is optimized without considering long-term workload balance or assignment equity. Our framework encourages sustainable and interpretable policies, avoiding systematic overuse of employees.

To comprehensively evaluate the agent's performance, we define three realistic operational scenarios: Base, Under-sampling, and Over-sampling, capturing various workforce-to-demand ratios. This multi-scenario evaluation framework allows us to assess the robustness and adaptability of the RL agent under diverse constraints, including resource scarcity and abundance.

Our contributions can be summarized as follows:

- We develop a realistic and flexible simulation environment for workforce allocation, modeling key operational variables.
- We design a reward function that integrates fairness criteria, ensuring balanced employee usage over time.
- We define and test the agent across three distinct workforce scenarios, demonstrating its adaptability and generalization capacity.

II. RELATED WORKS

A. Strategic Workforce Planning

Strategic Workforce Planning (SWP) is a continuous decision-making process that aligns an organization's human capital, its skills, capabilities, and workforce size, with long-term strategic goals and projected operational needs [18], [2]. Core activities include analyzing the existing workforce, forecasting supply and demand, identifying skill gaps, and implementing interventions such as hiring, training, or restructuring [14]. SWP requires continual adaptation to evolving organizational priorities and external market conditions [11], [3].

Historically, a range of quantitative and qualitative methods have been applied to SWP. Widely used quantitative

*All the authors are with the University of Padova, Department of Information Engineering, 35131 Padova (e-mail: valentina.tonazzo.1@studenti.unipd.it, marina.ceccon@phd.unipd.it, francesco.boscolomeneguolo@studenti.unipd.it, gianantonio.susto@unipd.it).

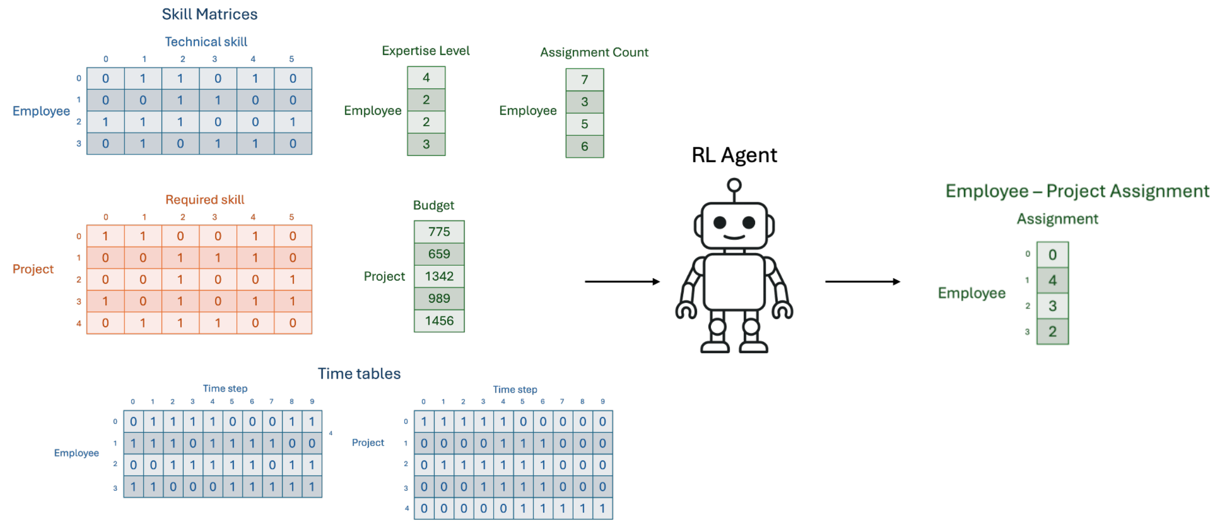


Fig. 1. Overview of the environment used to train the RL agent. Employees are represented by technical skill matrices, expertise levels, and historical assignment counts, while projects specify required skills, temporal demands, and budget constraints. Both employees and projects have associated time tables capturing dynamic availability. The RL agent learns to assign employees to projects under skill compatibility, economic efficiency, and workload fairness considerations.

techniques include heuristic methods and linear programming (LP). Heuristic methods rely on rule-based strategies to efficiently generate high-quality solutions for complex, large-scale problems where exact optimization is computationally infeasible [8].They offer flexibility and speed but lack global optimality guarantees and often require careful parameter tuning. By contrast, LP provides a formal framework for optimally allocating workforce resources by maximizing or minimizing a linear objective under linear constraints [6], [9], [15]. While LP models are rigorous and interpretable, they can struggle to accommodate non-linearities, uncertainty, and dynamic workforce behaviors [1].

Building on these foundations, more advanced techniques have emerged. Jaillet et al. [10] propose a robust optimization framework that models workforce uncertainty, including attrition, productivity, and managerial constraints, under a risk-averse objective. Network-based models [9] optimize long-term workforce transitions via linear programming, while dynamic programming approaches [15] capture hiring decisions over time. Recognizing workforce heterogeneity, Fowler et al. [8] introduce a mixed-integer programming model that incorporates cognitive ability as a worker attribute. Data-driven methods have also gained traction; for example, Eichenseer et al. [7] use machine learning to forecast short-term labor demand.

While these approaches rigorously address strategic planning objectives, they generally operate at an aggregate level. In contrast, our work focuses on the operational challenge of dynamically assigning individual employees to tasks based on fine-grained suitability metrics and time-dependent availability constraints.

B. Reinforcement Learning and Fairness

Reinforcement Learning (RL) offers a powerful paradigm for tackling complex sequential decision-making problems [19]. In RL, an agent learns to optimize its behavior through interactions with an environment, selecting actions based on observed states, receiving feedback in the form of rewards, and iteratively refining a policy to maximize long-term cumulative rewards.

Advances in deep reinforcement learning have further extended RL's applicability to large-scale, unified decision-making systems [12]. Within workforce management, prior research employing RL has primarily focused on aggregate-level planning. For instance, Smit et al. [17] use deep RL to guide long-term workforce composition. However, a gap remains in addressing real-time, fine-grained task assignments, where greater adaptability is needed at the individual task level.

Concurrently, an expanding line of research has explored how to embed fairness considerations into reinforcement learning reward structures [16], [4]. Nonetheless, to the best of our knowledge, these fairness-oriented methods have not yet been extended to operational-level task assignment within the context of Strategic Workforce Planning (SWP).

In this work, we address these gaps by proposing a novel RL-based approach for real-time operational decision-making in SWP. Specifically, we dynamically assign tasks based on individual suitability and availability, moving beyond aggregate resource allocation. A key innovation of our framework is integrating fairness constraints directly into the reward function, penalizing policies that exhibit disproportionate assignment patterns. This encourages the agent to jointly optimize for efficiency and fairness, preventing systematic overburdening of individual employees. Our approach thus facilitates robust, adaptive decision-making in dynamic and heterogeneous work environments.

III. OUR SETTING

We simulate a workforce management scenario, represented in figure 1, in which a Reinforcement Learning (RL) agent is trained to allocate employees to ongoing projects within an organization. The environment has been carefully crafted to reflect realistic operational constraints and dynamic conditions typically encountered in strategic workforce planning, especially in consulting firms. Several modeling decisions were made to balance complexity with computational tractability, ensuring that the agent is exposed to rich, structured observations while still allowing for efficient training and meaningful policy development.

A. Representation of Employees and Projects

Employees are represented by a synthetic binary skill matrix, where each row corresponds to an employee and each column represents a specific technical skill. A value of 1 indicates possession of a skill, while 0 indicates its absence. This abstraction allows the agent to reason systematically about skill matching. To capture variation in experience and seniority, each employee is also assigned a discrete expertise level, ranging from 1 (junior) to 4 (senior). Associated with each expertise level is a cost coefficient, reflecting both the employee's salary and their potential productivity impact. While real-world human capital includes a wide array of complex factors, such as soft skills, adaptability, motivation, and career aspirations, these aspects were deliberately abstracted to focus on the core technical matching process and to maintain a manageable state representation.

Projects are modeled similarly as synthetic binary matrices, where each row corresponds to a project and each column to a required skill. Each project specifies the technical competencies needed for successful completion, and projects may differ significantly in skill composition and intensity. Additionally, each project has associated temporal requirements, budget constraints, and resource demands, all of which the agent must take into account when making assignment decisions.

B. Temporal Dynamics and Workload Considerations

In order to model the dynamic nature of workforce planning, both employees and projects are assigned timeline vectors that evolve over discrete time steps. These binary vectors encode active (1) and inactive (0) statuses, allowing the agent to reason over availability, project durations, employee utilization, and scheduling constraints. Through these temporal features, the environment simulates real-world fluctuations in project demands and workforce availability, requiring the agent to plan not only for immediate assignments but also for future resource needs.

Moreover, to encourage fair and sustainable workforce management practices, the environment tracks each employee's assignment history. This record shapes the agent's policy, helping prevent over-reliance on specific individuals and fostering a more balanced workload distribution across the organization. By factoring assignment history into the state representation, the model supports policies that balance operational efficiency with employee well-being.

IV. CONSIDERED SCENARIOS

To assess the robustness and adaptability of the RL agent, we design three scenarios reflecting different levels of workforce availability relative to project demands. This is a key contribution of our work, as prior studies typically consider only one setting. In contrast, our framework evaluates agent behavior under balanced, constrained, and abundant resource conditions, enabling a more comprehensive test of its generalization capabilities.

In the **Base scenario**, a one-to-one correspondence exists between available employees and project requirements. This neutral setting ensures that every staffing need can be met without surplus or shortage, serving as a controlled benchmark for analyzing agent behavior in the absence of resource imbalances.

The **Under-sampling** scenario introduces a resource constrained environment, where the number of available employees falls short of project demands. This setting simulates realistic staffing shortages, requiring the agent to make prioritization decisions, accept suboptimal matches, and manage trade-offs. The reward function is adjusted to penalize both infeasible and inefficient assignments, encouraging effective resource allocation under pressure.

In contrast, the **Over-sampling** scenario models a surplus condition in which the workforce exceeds project needs. The agent is incentivized to avoid unnecessary or redundant assignments through reward penalties, promoting selective and strategic deployment of personnel to maximize efficiency and minimize underutilization.

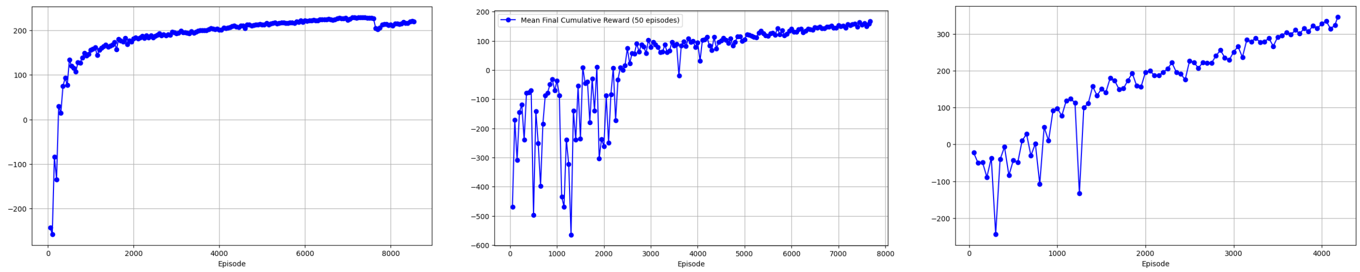
In all scenarios, the environment's step function updates the system state and calculates rewards based on the feasibility and efficiency of the agent's actions. This multi-scenario framework enables a systematic assessment of the agent's adaptability and performance in varied and dynamic workforce management settings.

V. METHODOLOGY

To address the workforce allocation problem, we built a custom Reinforcement Learning environment using the OpenAI Gym interface that simulates the complex dynamics of managing a company's human resources. It captures key employee–project interactions while embedding operational constraints and business objectives.

We train a Proximal Policy Optimization (PPO) agent, chosen for its stability in continuous control tasks. The agent is implemented using the `Stable-Baselines3` library, which integrates with the Gym environment and supports efficient training and experimentation.

The environment's state captures key components: the matrix of employee skills, the project requirements, the current allocations, the employee experience levels, the individual assignment counts, and the timestep indicator. This modular structure enables flexible, interpretable, and efficient handling of heterogeneous information.



Base Scenario Under-sampling Scenario Over-sampling Scenario

Fig. 2. Final reward per episode during training, averaged over a sliding window.

The **action space** is defined as a discrete set of all possible assignment pairs (i, j), representing the allocation of employee i to project j. This action formulation enables the agent to construct assignment plans incrementally, navigating a dynamically evolving environment characterized by resource availability and project requirements.

The cornerstone of the learning process is the design of a multi-objective **reward function**, aimed at reflecting key operational priorities:

$$r_i = \alpha + \beta S_i^M - \gamma c^e - P_i^e \tag{1}$$

where:

- S_i^M quantifies the skill match quality between the assigned employee and the target project role.
- c^e denotes the cost associated with assigning the employee, promoting cost-effective decisions.
- P_i^e introduces a penalty proportional to repeated assignments of the same employee, fostering equitable workload distribution.
- $\alpha > 0$ is a constant baseline reward to stabilize initial learning dynamics and encourage early exploration.

Fairness is explicitly embedded in the reward structure: the penalty term P_i^e increases with both the frequency and recency of assignments, discouraging over-utilization of specific employees and promoting a balanced distribution of tasks across the workforce. This design choice reflects ethical considerations often neglected in traditional RL-based resource allocation systems.

To improve realism and adaptability, the reward function is tailored to each operational scenario. In the **Under-sampling** case, where workforce is limited, the agent is penalized for suboptimal assignments, with penalties scaled to the severity of skill mismatches. In the **Over-sampling** case, a bonus is awarded upon first project completion, while idle employees incur penalties at episode end, encouraging efficient workforce use.

Scenario-specific step functions govern state transitions and rewards, reflecting unique constraints and incentives of each setting. Episodes terminate once all project roles are filled. In Under-sampling, episodes may also end early when no employees remain. In Over-sampling, termination occurs once staffing is complete, even if employees remain unassigned, triggering a penalty for unused capacity.

VI. Results

The PPO agent is trained separately in the three scenarios: **Base**, **Under-sampling**, and **Over-sampling**. Each training run consists of $N = 10000$ episodes, with a maximum of $T = 30000$ timesteps per episode. The availability of the employees and the duration of the projects are considered in a range of 30 days. In the various scenarios, a different number of employees and projects were used: 6 employees and 2 projects in the **Base**, 6 employees and 13 projects in the **Under-sampling**, 15 employees and 2 projects in the **Over-sampling**. For all scenarios, 3 skills and 4 employee experience levels were used.

The agent's performance is monitored through several metrics, including **cumulative reward** and **number of timesteps**. All experiments are conducted under identical hyperparameter settings with a learning rate of $L_R = 0.0003$, discount factor $\gamma = 0.99$, and clip range $= 0.2$. To support a detailed comparison between training and testing phases, we report the following metrics:

- $\mu_t \pm \sigma_t$: average number of timesteps per episode $[Ts]$,
- M_t: maximum number of timesteps $[Ts]$,
- m_t: minimum number of timesteps $[Ts]$,
- $\mu_r \pm \sigma_r$: average final cumulative reward in points $[P]$,
- M_r: maximum final cumulative reward achieved $[P]$.

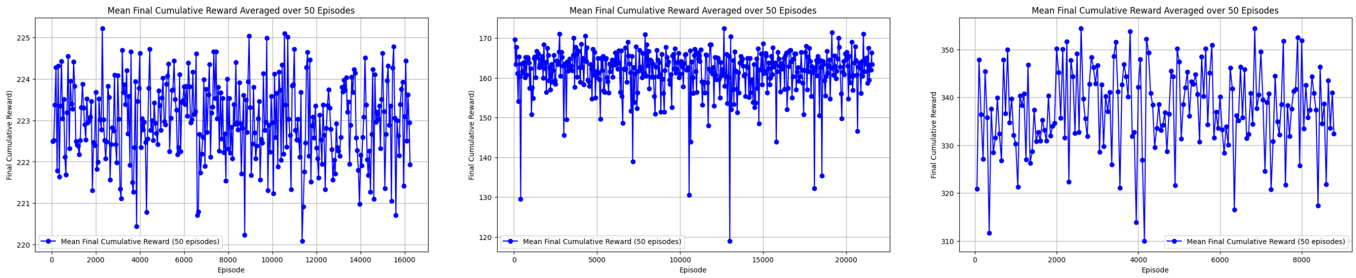
A. Learning Performance Analysis

Figure 2 shows the evolution of the final reward per episode during training, averaged over a sliding window.

In the **Base** scenario (Figure 2, left), the agent exhibits rapid performance gains within the first 2,000 episodes, followed by more gradual, linear improvements. This pattern suggests that the agent quickly learns an effective initial policy and incrementally refines it over time. A slight decline in performance around episode 4,000 may reflect renewed exploration or the onset of a learning plateau.

In the **Under-sampling** scenario (Figure 2, center), learning is slower and more erratic. The agent initially struggles to discover viable strategies, as indicated by the presence of strongly negative rewards. While performance improves over time, the learning curve remains unstable, highlighting the increased difficulty introduced by the limited availability of staff.

In the **Over-sampling** scenario (Figure 2, right), early training progress is slow and inconsistent. However, in



Base Scenario | Under-sampling Scenario | Over-sampling Scenario

Fig. 3. Final reward per episode during testing, averaged over a sliding window.

Phase	μ_t [Ts]	M_t [Ts]	m_t [Ts]	μ_r [P]	M_r [P]
Train	11.84 ± 2.02	24.20	10.10	183.54 ± 79.55	227.05
Test	13.82 ± 0.26	14.80	13.18	209.25 ± 3.52	215.25

TABLE I

COMPARISON OF EXPERIMENTAL RESULTS BETWEEN TRAINING AND TEST PHASES IN THE BASE SCENARIO.

Phase	μ_t [Ts]	M_t [Ts]	m_t [Ts]	μ_r [P]	M_r [P]
Train	13.24 ± 2.84	20.52	8.94	49.21 ± 116.89	165.37
Test	9.84 ± 1.79	14.08	8.38	166.49 ± 42.85	182.26

TABLE II

COMPARISON OF EXPERIMENTAL RESULTS BETWEEN TRAINING AND TEST PHASES IN THE UNDER-SAMPLING SCENARIO.

Phase	μ_t [Ts]	M_t [Ts]	m_t [Ts]	μ_r [P]	M_r [P]
Train	22.92 ± 2.34	28.64	19.12	176.61 ± 133.27	341.03
Test	21.06 ± 1.46	23.30	19.86	338.54 ± 7.65	356.79

TABLE III

COMPARISON OF EXPERIMENTAL RESULTS BETWEEN TRAINING AND TEST PHASES IN THE OVER-SAMPLING SCENARIO.

later stages, the agent achieves the highest observed rewards—stabilizing around 300 points—demonstrating its ability to exploit project completion bonuses while minimizing penalties for idle resources.

B. *Test Performance Analysis*

To assess the agent's generalization capability, a separate test phase was conducted for each scenario using the policies learned during training (Figure 3).

In the **Base** scenario (Figure 3, left), the agent consistently maintains high cumulative rewards throughout the test episodes, confirming the robustness of the policy learned under balanced conditions.

In the **Under-sampling** scenario (Figure 3, center), performance exhibits greater variability, consistent with the challenges posed by resource scarcity. Nonetheless, the agent achieves higher average rewards during testing than in training, indicating successful policy consolidation.

In the **Over-sampling** scenario (Figure 3, right), the agent outperforms the Base scenario, reaching average rewards above 300 points. This suggests that it effectively generalizes its strategy to handle the workforce surplus by leveraging bonuses while avoiding penalties.

C. *Overall Results of Training and Test Phases*

Tables I, II, and III provide a quantitative comparison of episode duration and cumulative rewards across training and testing phases.

In the **Base** scenario, the agent converges efficiently, with short and stable episode durations and consistently high cumulative rewards. This confirms the agent's ability to develop an effective policy under balanced conditions.

In the **Under-sampling** scenario, shorter episode durations are observed, especially during testing, as episodes frequently end early due to the lack of available employees. Although this results in lower average rewards compared to the Base case, the agent still demonstrates the ability to find viable allocation strategies in a more constrained setting. The increase in test-time rewards compared to training suggests improved policy consolidation once exploration is no longer necessary.

In the **Over-sampling** scenario, episodes run longer due to the larger decision space created by workforce surplus. Despite this added complexity, the agent achieves the highest test-time cumulative rewards, successfully exploiting project completion bonuses while minimizing idle-employee penalties.

Overall, the agent generalizes well across diverse operational settings. It adapts its strategy to varying levels of resource availability, consistently identifying feasible and fair allocation policies tailored to each scenario's specific constraints and opportunities.

VII. Conclusion and Future Work

In this work, we presented an RL-based framework for workforce allocation using a custom environment that emulates a consulting firm's operational constraints. By modeling employees with structured skills, seniority levels, availability, and costs, and by integrating a reward function balancing efficiency and fairness, we trained a PPO agent that learns robust, adaptive allocation strategies.

Experimental results across the Base, Under-sampling, and Over-sampling scenarios show that the agent learns to balance skill compatibility, cost management, and equitable workload distribution. Fairness constraints in the reward function were key to promoting balanced assignments, even under resource-scarce or resource-abundant conditions. The adaptive reward structure effectively guided the agent's behavior across different settings, supporting the approach's generalizability.

Nonetheless, several limitations remain. The environment abstracts certain human factors, such as individual preferences, team dynamics, and stakeholder feedback. Furthermore, while fairness is explicitly incorporated, the criteria are predefined and static, limiting flexibility in contexts with evolving notions of equity.

Future work will explore adaptive or learned fairness metrics, predictive policies for hiring and restructuring, and interactive human-in-the-loop evaluation. Extensions could also model employee learning, satisfaction, and turnover, enhancing both the realism and strategic value of the framework. Ultimately, this work lays a foundation for RL-based decision systems that combine technological rigor with operational relevance.

References

[1] A. Ben-Tal and A. Nemirovski, "Robust solutions of uncertain linear programs," *Operations Research Letters*, vol. 25, no. 1, pp. 1–13, 1999. [Online]. Available: https://doi.org/10.1016/S0167-6377(99)00016-4

[2] P. F. Boxall, J. Purcell, and P. M. Wright, *The Oxford handbook of human resource management*. Oxford university press, USA, 2007.

[3] J. L. Brewer and K. C. Dittman, *Methods of IT Project Management*, 4th ed. Purdue University Press, 2022.

[4] M. Cederle, L. V. Piron, M. Ceccon, F. Chiariotti, A. Fabris, M. Fabris, and G. A. Susto, "A fairness-oriented reinforcement learning approach for the operation and control of shared micromobility services," 2025. [Online]. Available: https://arxiv.org/abs/2403.15780

[5] A. Cotten, "Seven steps of effective workforce planning," 2007, accessed from IBM Center for the Business of Government.

[6] G. Dantzig, *Linear Programming and Extensions*. Princeton University Press, 1963.

[7] P. Eichenseer, L. Hans, and H. Winkler, "A data-driven machine learning model for forecasting delivery positions in logistics for workforce planning," *Supply Chain Analytics*, vol. 9, p. 100099, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2949863524000426

[8] J. W. Fowler, P. Wirojanagud, and E. S. Gel, "Heuristics for workforce planning with worker differences," *European Journal of Operational Research*, vol. 190, no. 3, pp. 724–740, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377221707006017

[9] C. Gaimon, S. Mohan, and G. L. Thompson, "A network approach to cohort personnel planning using cross-sectional data," *Regional Science and Urban Economics*, vol. 17, no. 1, pp. 125–139, 1987.

[10] P. Jaillet, G. G. Loke, and M. Sim, "Strategic workforce planning under uncertainty," *Oper. Res.*, vol. 70, no. 2, p. 1042–1065, Mar. 2022. [Online]. Available: https://doi.org/10.1287/opre.2021.2183

[11] F. Jaleh, P. Saadati, L. Husamaldin, and F. Aladesuru, "Data-driven strategicworkforce planning in uk smes: A comparative analysis," in *2024 International Conference on Decision Aid Sciences and Applications (DASA)*, 2024, pp. 1–7.

[12] J. Jiang, X. Liu, T. Ren, Q. Wang, Y. Zheng, Y. Du, Y. Peng, and C. Zhang, "Deep reinforcement learning for solving management problems: Towards a large management mode," *arXiv preprint arXiv:2403.00318*, 2024.

[13] N. Llort, A. Lusa, C. Martínez-Costa, and M. Mateo, "A decision support system and a mathematical model for strategic workforce planning in consultancies," *Flexible Services and Manufacturing Journal*, vol. 31, no. 2, pp. 497–523, 2019.

[14] J. Luy, G. Hiermann, and M. Schiffer, "Strategic workforce planning in crowdsourced delivery with hybrid driver fleets," 2023. [Online]. Available: https://arxiv.org/abs/2311.17935

[15] P. P. Rao, "A dynamic programming approach to determine optimal manpower recruitment policies," *Journal of the Operational Research Society*, vol. 41, no. 10, pp. 983–988, 1990.

[16] A. Reuel and D. Ma, "Fairness in reinforcement learning: A survey," 2024. [Online]. Available: https://arxiv.org/abs/2405.06909

[17] Y. Smit, F. den Hengst, S. Bhulai, and E. Mehdad, "Strategic workforce planning with deep reinforcement learning," in *Machine Learning, Optimization, and Data Science: 8th International Conference, LOD 2022, Certosa Di Pontignano, Italy, September 18–22, 2022, Revised Selected Papers, Part II*. Berlin, Heidelberg: Springer-Verlag, 2022, p. 108–122. [Online]. Available: https://doi.org/10.1007/978-3-031-25891-6_9

[18] R. Sparkman, *Strategic Workforce Planning: Developing Optimized Talent Strategies for Future Growth*, 1st ed. London, United Kingdom: Kogan Page, 2018, [Accessed 29 Nov 2023]. [Online]. Available: https://ebookcentral.proquest.com/lib/ulster/detail.action?docID=5231118

[19] R. S. Sutton, A. G. Barto *et al.*, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.