Multi Period Team Assignment Problem

Kasper Veje Jakobsen

Abstract— **This study tackles the Multi Period Team Assignment Problem (TAP), where individuals have evolving and partially observable preferences. We propose an integrated framework combining a Kalman Filter-based learner for dynamic preference estimation with strategic team assignment methods, including Random Assignment, Upper Confidence Bound (UCB), and Thompson Sampling. Simulations show that exact optimization methods are good in smaller settings but have scalability issues. The Kalman Filter performs well in stable environments but struggles with frequent resets. Additionally, controlling the degree of exploration in the assignment strategies is challenging. This research highlights the potential of integrating dynamic preference estimation with strategic optimization for effective team assignments.**

I. Introduction

In dynamic organizational environments, forming effective teams is essential for success and innovation. Both in corporate settings, academic research groups, project-based initiatives or sports, the ability to assemble good teams can significantly enhance performance outcomes. Such teams boost productivity and improve satisfaction while minimizing conflicts.

However, the process of assigning individuals to teams is complex. Each person brings unique preferences and interpersonal dynamics that influence team performance and overall effectiveness. Traditional team formation methods often fail to account for nuanced preferences, leading to bad team compositions and suboptimal performance. Additionally, individual preferences are frequently not fully observable and may change over time, which is further complicating the assignment process. Addressing these challenges requires advanced modeling approaches, such as dynamic programming, which can optimize team assignments across multiple periods while adapting to evolving preferences and limited information.

Note, that the problem tackled in this report, is inspired by personal previous work on Room Assignments in Danish Boarding Schools [3] [1] [2]. Although the assignment process in Room Assignments are more constrained, the overall work on the dynamics with preferences may be directly applicable in the room assignment process.

II. Problem Description

In this section, I will first give a high-level description of the Team Assignment Problem (TAP) and then introduce the appropriate mathematical notation for the problem. Furthermore, a more rigorous description of the problem is given.

The core problem involves optimally assigning a finite set of individuals to teams across multiple sequential periods, with the goal of maximizing overall team performance. Each individual has unique preferences toward every other individual, which influences team performance. However, these preferences are not directly observable. Only limited and noisy feedback is obtained based on current team assignments. Additionally, preferences evolve over time, adding further complexity to the assignment process. The goal is to maximize the overall performance across teams for all periods.

A. *Mathematical Notation*

Let $I = \{1, 2, \dots, n\}$ be the set of individuals, where $n < \infty$ represents the number of individuals. Let $P = \{1, 2, \dots, m\}$ be the set of sequential periods, where $m < \infty$ is the number of periods. Notice that the assumption that P is finite could be changed, if we consider an infinite number of periods. For each period $p \in P$, individuals are assigned to teams such that each individual belongs to exactly one team. A team assignment can be formalized as

$$\mathcal{T}^p = \{T_1^p, T_2^p, \dots, T_k^p\}$$

where k is the total number of teams at period p.

Each team T_j^p is a set of individuals defined as

$$T_j^p = \{i \in I \mid \text{Individual } i \text{ is in team } j \text{ at period } p\}$$

Assume that each individual belongs to exactly one team at each period and that each team has a maximum team size, i.e.,

$$\begin{aligned} &\bigcup_{j=1}^{k} T_j^p = I \\ \text{and} \quad &T_j^p \cap T_k^p = \emptyset, \quad \forall j, k, j \neq k \\ &|T_j^p| \leq \alpha, \quad \forall j, \forall p \in P \end{aligned}$$

Finally, we define

$$\mathcal{T} = \{\mathcal{T}^1, \mathcal{T}^2, \dots, \mathcal{T}^m\}$$

as the total set of team assignments in all periods.

Each individual has preferences towards all other individuals, which changes over time. Let w_{ij}^p denote the (true) preference of individual i towards individual j at period p, where w_{ij}^p is a value indicating the strength or quality of the preference. Let $w_{ij}^p \in \mathbb{R}$, where a negative value indicates a low/bad preference and a positive value indicates a strong/good preference. For simplicity, we define $W_p \in \mathbb{R}^{n \times n}$ as the matrix containing the preferences among the individuals at period p.

To introduce variability in the preferences, we let $\varepsilon_{ij}^p \sim N(0, \sigma_w^2)$ and define the following dynamics

$$w_{ij}^{p+1} = w_{ij}^p + \varepsilon_{ij}^p$$

At each period p, sparse feedback $F^p \in \mathbb{R}^{n \times n}$ is collected from individuals on their preferences towards their teammates. Specifically, F_{ij}^p represents the feedback provided by individual i about individual j and it is only available when both i and j are in the same team at period p.

$$F_{ij}^p = f(w_{ij}^p, \mathcal{T}^p)$$

where f is a feedback function. The feedback function f models how the true preferences and team dynamics influence the observed feedback. For example, f could be $f(w_{ij}^p, \mathcal{T}^p) = w_{ij}^p$ if feedback is unbiased. In this report, I let $\eta_{ij}^p \sim N(0, \sigma_f^2)$ and assume that

$$f(w_{ij}^p, \mathcal{T}^p) = \begin{cases} w_{ij}^p + \eta_{ij}^p & \text{if } i, j \text{ in same team at } p \\ \text{Not observed,} & \text{otherwise} \end{cases}$$

The performance of a team T_j^p can be modeled and calculated in many different ways. The following suggestions are custom-defined performance indicators (inspired by the L_1 and L_∞ measurements), that aggregate the pairwise preferences within a team. They are designed to reflect the overall compatibility and cohesion among team members. Let $L_\ell(T)$ be the ℓ measurement of the preferences of team T, such that

$$L_1^p(T) = \sum_{i,j \in T, i \neq j} w_{ij}^p$$
$$L_\infty^p(T) = \min_{i,j \in T, i \neq j} w_{ij}^p$$

Then, we can measure the total performance of all teams $\mathcal{T}^p$ in each period and over all periods as

$$L_\ell^p(\mathcal{T}^p) = \sum_{T \in \mathcal{T}^p} L_\ell^p(T), \quad \forall \ell \in \{1, \infty\}, p \in P$$
$$L_\ell(\mathcal{T}) = \sum_{p \in P} L_\ell^p(\mathcal{T}^p), \quad \forall \ell \in \{1, \infty\}$$

B. Extension of Preference Dynamics

To model real-world scenarios where individuals may enter or exit an organization continuously, we introduce a mechanism for resetting individuals with a probability γ at the end of each period. Formally, for each individual $i \in I$, a reset event occurs with probability γ after period p. When an individual is reset, their preferences towards all other individuals are reinitialized by drawing new values from the initial distribution. Mathematically, this can be expressed as:

$$w_{ij}^{p+1} = \begin{cases} \sim \mathcal{N}(\mu_0, \sigma_0^2) & \text{if } i \text{ is reset at } p, \\ w_{ij}^p + \varepsilon_{ij}^p & \text{otherwise,} \end{cases}$$

where $\mathcal{N}(\mu_0, \sigma_0^2)$ denotes the normal distribution with mean μ_0 and variance σ_0^2, and ε_{ij}^p represents the process noise as previously defined.

C. Comments

The goal of the problem is to maximize overall performance across all periods, which is a challenging task. Several problems must be addressed to achieve this objective. Firstly, individual preferences w_{ij}^p are only partially observable. This limited visibility makes it difficult to accurately predict team compatibility. Secondly, preferences evolve over time (personal growth or changing relationships), requiring the model to adapt to these dynamics. Additionally, the computational complexity increases with the number of individuals n and periods m. Finally, there is a need to balance exploration (trying different team configurations to learn more about preferences) with exploitation (using current knowledge to maximize team performance).

III. DP Model

In this section, the team assignment problem is modeled within the framework of dynamic programming. The goal is to devise an optimal strategy for assigning individuals to teams over multiple periods to maximize overall performance.

Consider a finite-horizon dynamic programming problem over periods $p = 1, 2, \ldots, m$. At each period p, the system is in a state s_p, an action a_p is chosen, leading to a reward r_p, and the system transitions to the next state s_{p+1}.

A. State Space

Since the system is partially observable, the true preferences w_{ij}^p are unobserved. The belief state at period p is the probability distribution over the possible values of w_{ij}^p, given all past observations and actions (history).

$$\hat{s}_p = \left\{ P(w_{ij}^p \mid \text{history up to } p) \mid i, j \in I \right\}.$$

The belief state summarizes the knowledge about the true preferences based on past team assignments and observed feedback and contains all the information necessary to make optimal decisions. Maintaining these probability distributions might however be problematic. Hence, we show that the belief state can be fully characterized by estimated preferences μ_{ij}^p and uncertainty σ_{ij}^p. That is, μ_{ij}^p and σ_{ij}^p are sufficient statistics for the belief state. The proof relies on the following assumptions, that are highlighted for clarity.

- The initial true preferences w_{ij}^1 are normally distributed.
- The evolution noise $\varepsilon_{ij}^p \sim N(0, \sigma_w^2)$ is normally distributed.
- The observation noise in feedback $\eta_{ij}^p \sim N(0, \sigma_f^2)$ is normally distributed.
- The true preferences evolve linearly with additive normal noise:

$$w_{ij}^{p+1} = w_{ij}^p + \varepsilon_{ij}^p$$

- The feedback (when i and j are in the same team) is a linear observation of the true preferences:

$$F_{ij}^p = w_{ij}^p + \eta_{ij}^p$$

Under these assumptions, we can show, that the posterior distribution $P(w_{ij}^p \mid \text{history})$ is normal with mean μ_{ij}^p

and variance σ_{ij}^p, and these parameters can be recursively updated.

In the first period, we have that $P(w_{ij}^1) = N(\mu_{ij}^1, \sigma_{ij}^1)$. Then the following recursive update from period p to $p+1$ can be used.

$$\begin{aligned} &P(w_{ij}^{p+1} \mid \text{history up to } p) \\ &= \int P(w_{ij}^{p+1} \mid w_{ij}^p) P(w_{ij}^p \mid \text{history up to } p) \, \mathrm{d}w_{ij}^p \end{aligned}$$

Since both w_{ij}^p and ε_{ij}^p are normal, w_{ij}^{p+1} is also normal with

$$\mu_{ij}^{p+1|p} = \mu_{ij}^p, \quad \sigma_{ij}^{p+1|p} = \sigma_{ij}^p + \sigma_w^2.$$

If feedback F_{ij}^{p+1} is observed, then we can use Bayes' theorem to deduce that

$$\begin{aligned} &P(w_{ij}^{p+1} \mid \text{history up to } p+1) \propto \\ &P(F_{ij}^{p+1} \mid w_{ij}^{p+1}) P(w_{ij}^{p+1} \mid \text{history up to } p) \end{aligned}$$

Since both the prior and likelihood are normal, the posterior is normal with

$$\mu_{ij}^{p+1} = \mu_{ij}^{p+1|p} + K_{ij}^{p+1}\left(F_{ij}^{p+1} - \mu_{ij}^{p+1|p}\right),$$

$$\sigma_{ij}^{p+1} = \left(1 - K_{ij}^{p+1}\right)\sigma_{ij}^{p+1|p},$$

where the Kalman gain K_{ij}^{p+1} is

$$K_{ij}^{p+1} = \frac{\sigma_{ij}^{p+1|p}}{\sigma_{ij}^{p+1|p} + \sigma_f^2}$$

If the feedback, is not observed, we simply just have that

$$\mu_{ij}^{p+1} = \mu_{ij}^{p+1|p}, \quad \sigma_{ij}^{p+1} = \sigma_{ij}^{p+1|p}$$

Hence, the posterior distribution $P(w_{ij}^p \mid \text{history})$ remains normal at each period, and it is fully characterized by μ_{ij}^p and σ_{ij}^p. These parameters summarize all the necessary information from the history, making them sufficient statistics for the belief state. Furthermore, we can directly apply a Kalman Filter approach to directly model and update the sufficient statistics based on team interactions.

To furthermore, encapsulate the extension on preference dynamics involving resetting preferences for certain individuals with a probability γ, we can at each period p also reset the estimates of the preferences for these individuals.

Finally, the sufficient belief state s_p at period p encapsulates all the information necessary to make optimal decisions moving forward.

$$s_p = \{\mu^p, \sigma^p\}$$

Where $\mu^p \in \mathbb{R}^{n \times n}$, such that μ_{ij}^p is the estimated preferences between individuals i and j. Furthermore, $\sigma^p \in \mathbb{R}^{n \times n}$, such that σ_{ij}^p is the uncertainty (variance) associated with μ_{ij}^p. The sufficient belief state space is continuous and infinitely large.

B. Action Space

The action a_p at period p is the team assignment $\mathcal{T}^p$. The set of feasible actions $A(s_p)$ consists of all possible partitions of the individual set I into disjoint teams that satisfy problem constraints (maximum team size and number of teams). Thus:

$$a_p = \mathcal{T}^p \in A(s_p)$$

C. Transition Function

The transition from s_p to s_{p+1} can be determined using the Kalman filter approach described in the proof of the sufficient statistic. This procedure is encapsulated by f, where F^p is the feedback given on action a_p at period p.

$$s_{p+1} = f(s_p, a_p, F^p)$$

Notice, that if σ_w^2 and σ_f^2 are not known, we need to approximate both parameters using observed data during this procedure.

D. Reward Function

The reward r_p at period p is the total performance of all teams. This was initially defined using the true preferences, but can be approximated using the estimated preferences μ_{ij}^p and the L_1 measure, such that

$$r_p = \sum_{T \in \mathcal{T}^p} \sum_{i,j \in T, i \neq j} \mu_{ij}^p$$

E. Objective Function

The objective is to find a policy $\pi = \{\pi_1, \pi_2, \ldots, \pi_m\}$ that specifies the action a_p at each period p to maximize the expected total reward:

$$\max_\pi E^\pi \left[\sum_{p=1}^m r_p\right].$$

F. Policy and Value Functions

Define the value function $V_p(s_p)$ as the maximum expected total reward from period p onward, given state s_p:

$$V_p(s_p) = \max_{a_p \in A(s_p)} \left[r_p + E\left[V_{p+1}(s_{p+1}) \mid s_p, a_p\right]\right]$$

with terminal condition $V_{m+1}(s_{m+1}) = 0$.

The optimal policy π^* satisfies the Bellman optimality equation:

$$\pi_p^*(s_p) = \arg\max_{a_p \in A(s_p)} \left[r_p + E\left[V_{p+1}(s_{p+1}) \mid s_p, a_p\right]\right].$$

IV. Solution Approaches

Addressing the Team Assignment Problem requires strategies that can handle the complexities of a continuous and expansive state and action space. Traditional DP methods become impractical due to the infinite and partially observable state space and the combinatorial explosion of possible team assignments. Therefore, we adopt an Approximate DP framework utilizing Upper Confidence Bound (UCB) and Thompson Sampling strategies, complemented by Integer Programming (IP) to optimize team configurations at each decision stage. Additionally, a Random Assignment approach is formalized as a baseline for performance comparison.

Note, that direct application of the standard DP value function approaches to TAP is hindered by two primary challenges:

- Continuous and Infinite State Space: The belief state, characterized by estimated preferences μ_{ij}^p and uncertainties σ_{ij}^p, forms a continuous and infinitely large state space.
- Exponential Action Space: The number of possible team assignments grows exponentially with the number of individuals n. This large action space makes it impractical to evaluate all possible actions at each decision point using exact methods.

A. Random Assignment

The Random Assignment approach serves as a baseline method, where individuals are allocated to teams without considering their estimated preferences or historical feedback. This stochastic strategy provides a reference point against which the performance of the following algorithms can be measured.

Using this approach, individuals are allocated to teams by uniformly sampling from all feasible team configurations. Specifically, given n individuals and k teams with a maximum size s, the method constructs a multiset $A = \{1, \ldots, 1, 2, \ldots, 2, \ldots, k, \ldots, k\}$ where each team index is repeated s times. A random permutation π of A is generated uniformly, and the first n elements of π are assigned to the individuals. Formally, the assignment function $a : I \to T$ is defined as $a(i) = \pi(i)$ for $i = 1, \ldots, n$. This ensures that each team receives a random subset of individuals while enforcing the maximum team size constraint.

B. Upper Confidence Bound (UCB) Strategy

The Upper Confidence Bound (UCB) strategy integrates both the estimated mean preferences and the associated uncertainties to compute team assignments. This strategy therefore balances exploration and exploitation. For each pair of individuals i and j at period p, a score S_{ij}^p is computed as follows

$$S_{ij}^p = \mu_{ij}^p + \beta \cdot \sigma_{ij}^p,$$

Note that $\beta \geq 0$ is a hyperparameter that controls the degree of exploration. A higher β value increases the influence of uncertainty, such that the assignment explore uncertain relations, while a lower β emphasizes exploitation of known preferences.

The team assignment a_p for period p is determined by solving the optimization problem:

$$a_p = \arg\max_{a \in \mathcal{A}} \sum_{T \in a} \sum_{\substack{i,j \in T \\ i \neq j}} S_{ij}^p$$

This objective function seeks to maximize the aggregate score across all teams, prioritizing pairings with high estimated preferences and significant uncertainty.

To solve the optimization problem, Integer Programming (IP) is used at each time period. The IP formulation encapsulates the constraints of the team assignment problem, such as maximum team sizes and the exclusivity of individual assignments, while optimizing the sum of the scores. Specifically, binary variables are defined to indicate the assignment of individuals to teams, and the IP solver identifies the team configuration that maximizes the total UCB score, which can be formalized as

$$\begin{aligned}
\text{Maximize} \quad & \sum_{t=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{n} S_{i,j}^p \cdot x_{i,t} \cdot x_{j,t} \\
\text{Subject to} \quad & \sum_{t=1}^{k} x_{i,t} = 1 \quad \forall i \in I, \\
& \sum_{i=1}^{n} x_{i,t} \leq \alpha \quad \forall t \in T, \\
& x_{i,t} \in \{0, 1\} \quad \forall i \in I, \forall t \in T,
\end{aligned}$$

where:

- $S_{i,j}$ is the score between individuals i and j.
- $x_{i,t}$ is a binary variable indicating whether individual i is assigned to team t.
- n is the total number of individuals.
- k is the number of teams.
- α denotes the maximum team size.
- I represents the set of all individuals.
- T represents the set of all teams.

By using the UCB strategy, the assignments optimizes immediate team performance based on current preference estimates but also uses the information on uncertainty to consider future team assignments. This should enhance overall performance in the problem.

C. Thompson Sampling Strategy

The Thompson Sampling strategy adopts a Bayesian approach to balance exploration and exploitation in the Team Assignment Problem (TAP). By sampling preferences based on the estimated mean and variance, this method adapts team assignments based on the underlying uncertainty in individual preferences, such that both immediate performance and exploration is balanced.

For each pair of individuals i and j at period p, Thompson Sampling generates a stochastic score S_{ij}^p by sampling from the posterior distribution of the preference estimates:

$$S_{ij}^p \sim \mathcal{N}\left(\mu_{ij}^p, \sigma_{ij}^p\right),$$

This sampling process incorporates both the estimated preferences and their uncertainties, allowing the both exploration of uncertain pairings while exploiting high preference estimates with low uncertainty.

Like in the UCB strategy, the team assignment a_p for period p is determined by solving the following optimization problem:

$$a_p = \arg\max_{a \in \mathcal{A}} \sum_{T \in a} \sum_{\substack{i,j \in T \\ i \neq j}} S_{ij}^p,$$

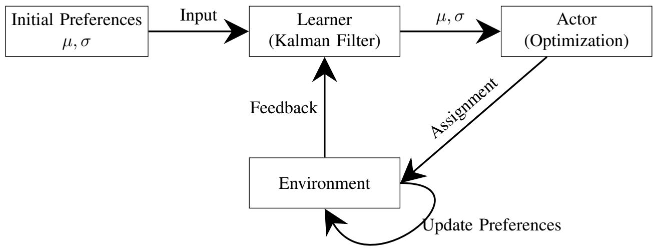

Fig. 1. Integration of Learner, Actor, and Feedback Loop with Preference Update in the Environment.

To solve the problem with the sampled preferences, Integer Programming (IP) is once again used to solve the optimization problem at each period. The IP formulation is similar to the one used in the UCB strategy, but naturally the score $S^p_{i,j}$ is now sampled as described above.

While Thompson sampling assembles the UCB strategy, the probabilistic nature of the sampling could provide an advantage in certain systems.

V. Integrated Framework

The Integrated Framework combines the interaction between preference estimation, team assignment strategies, and the dynamic environment to effectively solve the Team Assignment Problem. This framework is composed of three primary components: the *learner*, the *actor*, and the *environment*, as depicted in Figure 1.

A. Learner

At the core of the framework is the learner component, which uses a Kalman Filter to dynamically estimate the mean preferences μ^p_{ij} and their associated uncertainties σ^p_{ij} for each pair of individuals i and j at period p. The Kalman Filter recursively updates these estimates based on incoming feedback from the environment. The learner uses the approach outlined in subsection III-A.

Another important aspect of the learner's functionality is the estimation of the system variances σ_f and σ_w, which represent the observation noise and the process noise, respectively. Accurate estimation of these variances is essential for the Kalman Filter to appropriately weigh new observations against prior estimates. Based on the observed feedback, the learner uses the following methods, to estimate the system variances.

Estimation of σ_f: The observation noise variance σ_f is estimated by analyzing the residuals between the observed feedback F^p_{ij} and the predicted mean preferences μ^p_{ij}. Specifically, for each observed feedback, the residual is computed as:

$$r^p_{ij} = F^p_{ij} - \mu^p_{ij}$$

The variance of these residuals across multiple observations is then calculated to update σ_f:

$$\hat{\sigma}^2_f = \frac{1}{N-1} \sum_{p=1}^{N} \left(r^p_{ij} \right)^2,$$

where N is the number of observed feedback instances. To ensure numerical stability and non-negativity, the estimated variance is constrained as:

$$\hat{\sigma}^2_f = \max\left(\hat{\sigma}^2_f, 0\right)$$

Estimation of σ_w: The process noise variance σ_w captures the uncertainty in the evolution of preferences over time. It is estimated by evaluating the changes in the mean preference estimates μ^p_{ij} across consecutive periods:

$$\Delta\mu^p_{ij} = \mu^{p+1}_{ij} - \mu^p_{ij}$$

The variance of these changes is then computed as:

$$\hat{\sigma}^2_w = \frac{1}{M-1} \sum_{p=1}^{M} \left(\Delta\mu^p_{ij} \right)^2,$$

where M is the number of periods considered. Similar to σ_f, the estimated process noise variance is constrained to be non-negative:

$$\hat{\sigma}^2_w = \max\left(\hat{\sigma}^2_w, 0\right)$$

These adaptive estimations of σ_f and σ_w allow the learner to correctly update the preference estimates. They are directly applied in the equations used in the Kalman Filter.

Furthermore, the learner handles resetting the preferences of the individuals, if a signal is given from the environment. Naturally, the estimates are reset, but also the statistics computed to estimate the system variances has to be handled carefully.

B. Actor

The actor component uses the estimated preferences and uncertainties provided by the learner to formulate team assignments. The three different proposed assignment strategies can be used interchangeably in the actor.

C. Environment

The environment handles the system in which team assignments are executed. After receiving a team assignment a_p from the actor, the environment generates feedback F^p based on the interactions and actual preferences. This feedback is then fed back into the learner, enabling the Kalman Filter to update the preference estimates and system variances for the following periods.

Furthermore, the environment handles updating the true preferences including resetting certain individuals preferences.

D. Implementation

The implementation of the Integrated Framework for the Team Assignment Problem (TAP) is made through a modular and efficient Python codebase.

Python's `NumPy` library is used to perform efficient matrix computations, which are fundamental several components of the problem. Random data generation is handled using `NumPy`'s random module, which allows for the creation of synthetic datasets that simulate scenarios.

Furthermore, Gurobi has been invoked as the IP solver for the periodic team assignments for the UCB and Thompson Sampling strategies.

All source code for the Integrated Framework, including the environment simulations, assignment strategies (Random, UCB, Thompson Sampling), and the learner's Kalman Filter implementation, is publicly available at https://github.com/kveje/ISYE723-TAP.

VI. Results

This section presents the results of the simulation experiments conducted to evaluate the performance of the proposed solution approaches under varying conditions. The impact of different reset probabilities (γ) and different assignment strategies on team performance, preference estimation accuracy, and computational efficiency. The experiments were carried out with the following parameters: $n = 10$ individuals, $|T| = 4$ teams, a maximum team size of 3, $m = 100$ periods, and system variances $\sigma_w = 0.1$ and $\sigma_f = 0.1$. For each combination of agent and reset probability, the simulation was replicated 10 times to ensure statistical reliability, and the results were aggregated accordingly to show both mean and variance.

A. No Resets

To investigate how well the agents perform, without the reset disturbance of the preferences, we will start by presenting the results for $\gamma = 0$. This will act as a baseline, to understand how an increasing γ-value will affect the performance and preference estimation.

Figure 2 and Figure 3 illustrates the distribution of rewards obtained across different assignment strategies (actors). The rewards reflect the immediate performance of the team assignments based on the selected assignment strategy.

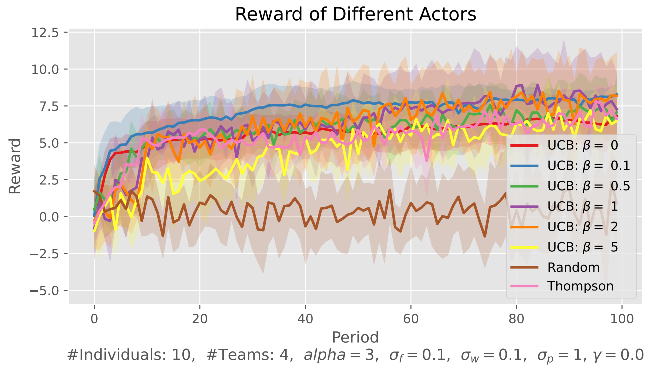

Fig. 2. Distribution of Rewards across Different Assignment Strategies

As expected, the random assignment baseline is outperformed by both the UCB strategies with varying β-parameters and the Thompson Sampling strategy. Additionally, the Thompson Sampling strategy is only better than the UCB strategy with $\beta = 5$, but it's cumulative reward of all periods are quite close to UCB with $\beta \in \{0, 0.5, 1, 2\}$. The best performing assignment strategy is UCB with $\beta = 0.1$, suggesting that a consideration to the variance promotes the long-term team performances. However both too high and too little incorporation of the variance gives worse results, suggesting that β might be a difficult hyperparameter to determine in global settings.

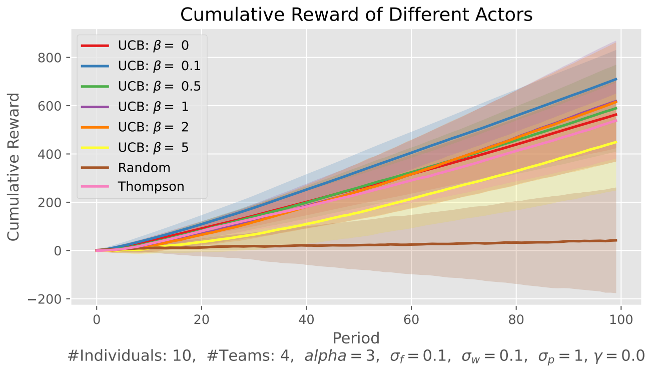

Fig. 3. Distribution of Cumulative Rewards across Different Assignment Strategies

We can also measure the performance of each assignment strategy, by evaluating the distance between the estimated preferences and the actual preferences. The preference distance is measured using the L^2-norm and Figure 4 presents the average preference distance and the associated variance for each agent.

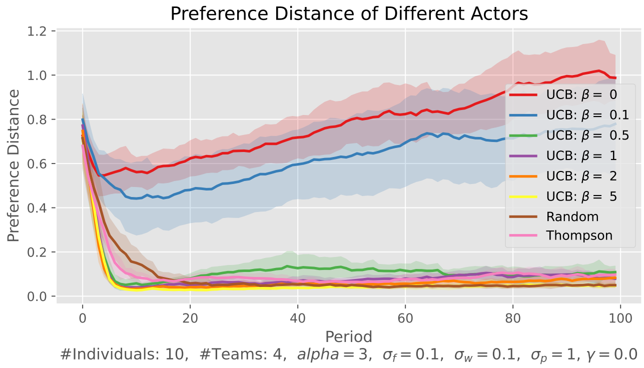

Fig. 4. Distribution of Preference Distance for Different Assignment Strategies

Lower preference distances indicate more accurate preference estimations. Only the two UCB strategies with $\beta \in \{0, 0.1\}$ does not show convergence of the preference estimations. This indicates, that exploiting the estimated preferences without considering the variance associated with the estimation results in non-converging preference estimations. Since both these strategies provide good performance measured on the rewards (UCB with $\beta = 0.1$ is the best strategy), the results on the preference distances suggests that some degree of uncertainty in the estimation might not be too worsening for the assignment strategy. All other strategies than the two mentioned above show significant convergence of preference estimates after only 20 periods, and UCB with $\beta \in \{0.5, 1, 2, 5\}$ converges after only 10 periods.

Efficiency is critical for practical applications. Figure 5 shows the average computational time per period for each agent under varying reset probabilities.

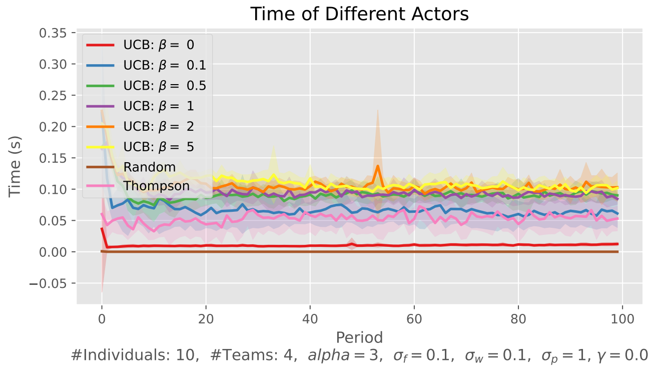

Fig. 5. Average Computational Time per Period for Different Assignment Strategies

The computational time remains relatively stable across the periods for all agents. The random strategy is by far the fastest assignment strategy taking less than 0.01 second per assignment, and solving the IP's for the UCB and Thompson Sampling strategy varies within 0.02 and 0.15 seconds.

B. Impact of Reset Probability, γ

To investigate the impact of changing the reset probability γ, we will present results from different agents with varying parameters of $\gamma \in \{0, 0.01, 0.05, 0.1\}$ with the same setup as in the previous section. 10 simulations are made, and the collected data is aggregated accordingly. We have already established, that the random strategy is easily outperformed by both the UCB and Thompson Sampling strategy, hence it will be left out.

First, we compare the Thompson Sampling strategy across the different levels of γ. The cumulative rewards are shown in Figure 6 and the preference distances are shown in Figure 7.

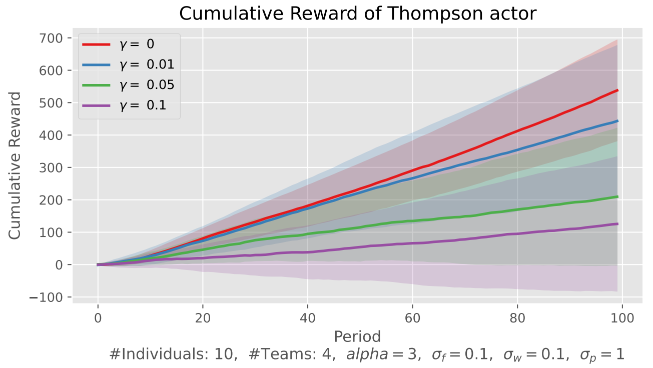

Fig. 6. Distribution of Cumulative Rewards across Different γ-values

Clearly, the team performance is directly affected by increasing the probability of an individual being reset. As γ increases, the cumulative reward significantly decreases. The reason might be directly related to the fact, that the estimated preferences does not converge to the true preferences due to the significant resets of individuals.

Similar results are computed for all other actors including UCB with different levels of β. This hints, that while the Kalman Filter approach works great when $\gamma = 0$, further methods might need to be developed to handle $\gamma > 0$.

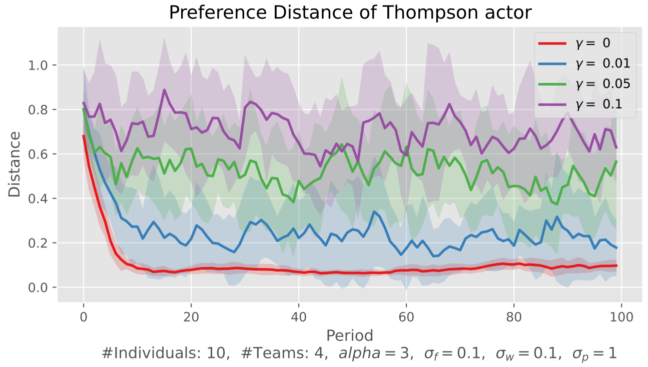

Fig. 7. Distribution of Preference Distance for Different γ-values

C. Increasing the Size of the System

To test the scalability of the approaches, a new set of experiments has been carried out with the following paramters: $n = 20$ individuals, $|T| = 6$ teams, a maximum team size of 4, $m = 100$ periods, and system variances $\sigma_w = 0.1$ and $\sigma_f = 0.1$. As previous, the simulation was replicated 10 times for each agent to ensure statistical reliability, and the results were aggregated accordingly to show both mean and variance. Note, that $\gamma = 0$ has been fixed.

By increasing the size of the system, the computational complexity of carrying out especially the IP's corresponding to solving the period-specific assignment problems increases. Hence, only the random, Thompson Sampling and UCB with $\beta \in \{0, 0.1, 0.5\}$ strategies have been considered.

First, the average computational time is shown in Figure 8. The figure clearly shows, that the time of each assignment has significantly increase for the Thompson Sampling strategy and the UCB with $\beta \in \{0.1, 0.5\}$. A small caveat is, that we had to introduce a hard time cap on 10 seconds for the UCB strategy with $\beta = 0.5$, since some assignments were not being solved within a minute. This shows that the exact solution approach (IP) might be infeasible for larger settings, especially in simulation studies.

Additionally, we can investigate the performance of each actor in the system. The cumulative rewards are shown in Figure 9 and the preference distance are shown in Figure 10. Both plots show a similar pattern than previous results from subsection VI-A. The UCB strategy with $\beta = 0.1$ is the best strategy, even though the preference estimates does not converge to the actual preferences - suggesting that exploiting known preferences instead of learning some uncertain preferences is beneficial. The Thompson Sampling strategy also seem to offer a good trade-off between exploration and exploitation, since the preferences quite quickly converge

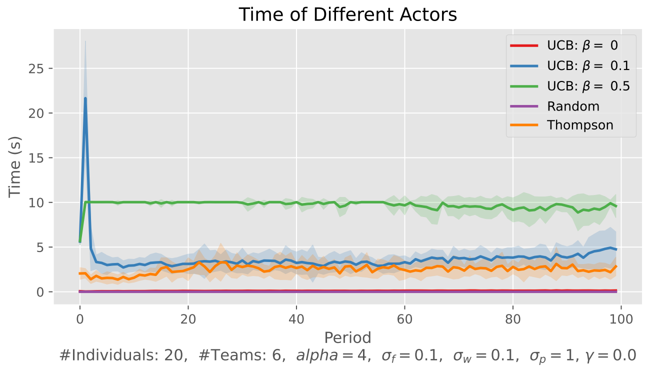

Fig. 8. Average Computational Time per Period for Different Assignment Strategies

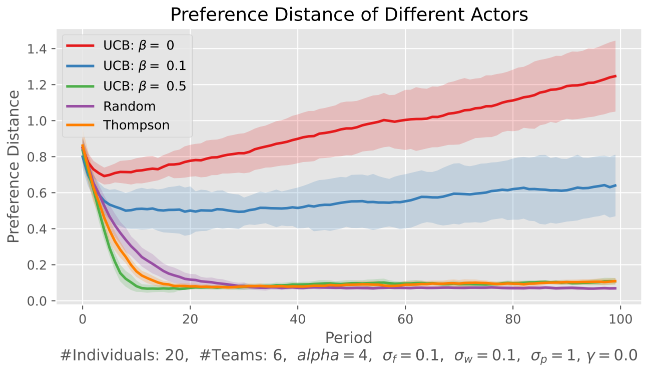

Fig. 10. Distribution of Preference Distance across Different Assignment Strategies

and the cumulative reward is almost on level with the best strategy.

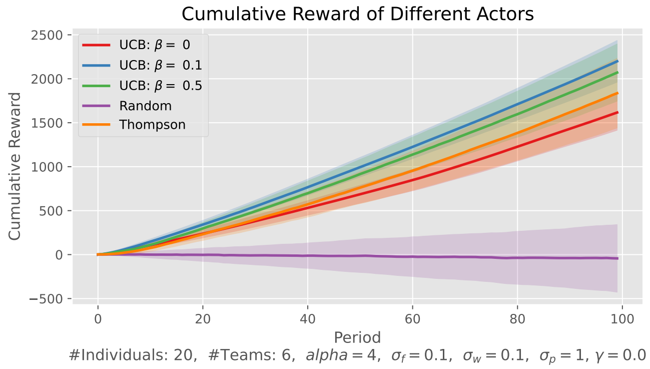

Fig. 9. Distribution of Cumulative Rewards across Different Assignment Strategies

VII. Discussion

The discussion addresses the key findings of the simulation results, highlighting the strengths and limitations of the investigated methods. Additionally, further improvements of the model will be suggested.

A. Integer Programming for Period-Specific Assignments

Using Integer Programming (IP) to determine the period-specific team assignments has proven effective in achieving optimal configurations, particularly in smaller-scale settings ($n = 10$, $|T| = 4$). However, the scalability of IP is a significant limitation when used in larger systems. As demonstrated in the experiments with $n = 20$ individuals and $|T| = 6$ teams, the computational time required to solve the IP formulations increases substantially from 0.10 seconds to $5 - 10$ seconds per period. The need to impose a hard time cap for the UCB strategy with $\beta = 0.5$ also shows the impracticality of IP in large-scale applications.

This limitation could be solved by using heuristic or approximate optimization methods, that offer a faster alternative for larger systems. Using such methods could provide near-optimal solutions with significantly reduced computational time. Integrating these approaches within the actor component could strengthen the framework's scalability, enabling it to handle larger team assignment scenarios without compromising performance. Also other assignment strategies, such as multi-agent approaches could be explored.

B. Effectiveness of the Kalman Filter Approach

The Kalman Filter-based learner demonstrates robust performance in scenarios without individual resets ($\gamma = 0$), effectively converging the estimates to the true preferences for certain assignment strategies. The filter's ability to continuously update preference estimates based on incoming feedback ensures that the team assignments are relying on the the underlying preferences.

However, the introduction of individual resets ($\gamma > 0$) poses a challenge to the Kalman Filter's effectiveness. The reinitialization of preferences is a major disturbance in the filter's convergence, leading to increased preference distances and diminished cumulative rewards. This indicates that while the Kalman Filter works great in stable environments, additional modifications might be necessary to handle dynamic changes. Potential solutions could involve mechanisms with adaptive learning rates or combining the Kalman Filter with other learning techniques to better handle the major changes in the preferences.

C. Choosing the Exploration Parameter β

The exploration parameter β has been shown to be very important when balancing exploration and exploitation within the UCB strategy. The simulation results reveal that $\beta = 0.1$ yields the best performance, suggesting that some emphasis (but not too much) on uncertainty provides the optimal long-term team performance. However, the sensitivity of the system to the value of β indicates that selecting an appropriate value is difficult. Both excessively high and low β values result in suboptimal outcomes.

Determining an optimal β in a global setting is challenging, since the dynamic nature of individual preferences vary across different periods. A potential approach to handle this issue is implementing a dynamic β that adapts based on the feedback and the rewards. For example, β could be adjusted according to the rate of convergence of preference estimates or the variance of the estimated preferences. Alternatively, Bayesian optimization techniques could be employed to dynamically adjust β in response to observed performance.

D. *Implications and Future Work*

The integrated framework demonstrates its ability to handle the Team Assignment Problem through a combination of dynamic preference estimation and strategic team assignment. The effective use of the Kalman Filter and different optimization strategies show the capability to enhance team performance and preference estimation accuracy in stable environments.

However, the challenges identified, particularly the scalability of IP, the Kalman Filter's limitations in dynamic settings, and the sensitivity to the exploration parameter β, highlight potentials for future research and development. Exploring different optimization methods, improving the learner's adaptability to dynamic changes, and developing mechanisms for parameter tuning is important for extending the framework's ability to handle more complex and realistic scenarios.

Additionally, incorporating other real-world factors such as heterogeneous team roles, varying individual capacities, and more sophisticated preference dynamics could further enhance the framework. Future studies could also investigate the integration of machine learning techniques to predict and adapt to individual resets.

While the current implementation of the integrated framework show strong capabilities, addressing the limitations through some of the suggested enhancements could be essential for its successful development.

VIII. Conclusion

This project addresses the complex challenge of optimally assigning individuals to teams in dynamic organizational environments, where preferences are both partially observable and subject to change over time. By formulating the Team Assignment Problem within a dynamic programming framework, we have developed an integrated framework that combines advanced preference estimation with strategic team formation strategies.

The integrated frameworks implements a Kalman Filter-based learner, which estimates mean preferences and their associated uncertainties, adapting to incoming feedback. The learner is paired with different assignment strategies, including Random Assignment, Upper Confidence Bound (UCB), and Thompson Sampling, leveraging Integer Programming (IP) to determine optimal team configurations at each period. The simulations, using different reset probabilities (γ) and system sizes, shows the potentials of these strategies in balancing exploration and exploitation to maximize team performance.

The results indicate that while IP is effective for smaller-scale systems, the scalability becomes a limiting factor as the system size increases, suggesting a need for alternative optimization methods in larger settings. Additionally, the Kalman Filter approach show good performance in stable environments ($\gamma = 0$), but its performance decreases with higher reset probabilities, highlighting the need for other learning mechanisms to handle dynamic changes. The sensitivity of the exploration parameter β highlights the challenge in parameter tuning, suggesting that adaptive or dynamically determined strategies could be developed.

In conclusion, the integrated framework successfully demonstrates significant results in team performance and preference estimation accuracy in controlled environments. Future work should focus on integrating heuristic optimization methods, improving the learner's adaptability through more advanced models, and developing additional mechanisms for dynamic parameter tuning.

References

[1] Kasper Veje Jakobsen. Answers to obligatory assignment 3, autumn 2023, 2023. Available at https://www.dropbox.com/scl/fi/hzlyeiq9kasdan8znaxhh/asg3.pdf?rlkey=vfn3zll9ac517b6vs4trfeoqs&st=qdtckvfk&dl=0.

[2] Kasper Veje Jakobsen. Answers to obligatory assignment 5, autumn 2023, 2023. Available at https://www.dropbox.com/scl/fi/yp01g4kjpmx9idwnyup6n/asg5.pdf?rlkey=rrq26rx61oom5qfrasacf4a6j&st=0q835gaf&dl=0.

[3] Kasper Veje Jakobsen. Bachelor report, 2023. Available at https://www.dropbox.com/scl/fi/b4nkqeqpu3oshg9usa72p/Bachelor_Report.pdf?rlkey=3e51gmzxm1b1kksyvolxuw756&st=ayd2hk6o&dl=0.