Original Article

Maximizing Customer Retention in Multi-session Training Service: Model and Algorithm

Production and Operations Management
1–20

DOI: 10.1177/10591478251369160
journals.sagepub.com/home/pao

Qiuwei Guo[1], Yifu Li[1], Lindong Liu[1] and Lifei Sheng[2]

Abstract

Training is an important business in the service sector. Usually, a training program involves multiple sessions and each session contains multiple activities. Although it is essential for customers to participate in all training sessions and activities, many customers fail to complete the program because the training experience is too stressful. Given the importance of customer retention in the training programs, we investigate the modeling and optimization of the retention-oriented training program design problem (RTDP), which maximizes overall service retention across all training sessions through activity scheduling. Customers make their participation decisions about their next training session based on the remembered holistic utility of past training activities. By our analysis, RTDP is a 0–1 constrained exponential sum problem, which we prove to be NP-hard. To resolve RTDP, we introduce a geometric branch and bound algorithm that efficiently searches for the optimal solution by resolving a series of subproblems. From a numerical study, we find that higher reward, difficulty, and value lead to more U-shaped, inverted U-shaped, and increasing subsequences in each session, respectively. The reason is that higher reward favors sequences with a pleasant start and a sharp positive gradient toward the end, higher difficulty requires warm-up and cool-down, and higher value makes customers emphasize the end experience. Finally, we extend our research by investigating RTDP with session breaks and discussing the joint retention-performance optimization. When there are breaks between sessions, we find that as the break duration increases, the optimal value first increases and then decreases. For the joint retention-performance optimization, the optimal sequence is more pulsed if the service designer cares more about customer performance and flatter if the service designer cares more about customer retention.

Keywords

Service Scheduling, Interactive Service Design, Customer Retention Optimization, Geometric Branch and Bound Algorithm, 0–1 Constrained Exponential Sum Problem

Date received 26 August 2024; accepted 30 July 2025 after two revisions

Handling Editor: Michael Pinedo

1 Introduction

Training is an essential service that affects everyone's daily life. Consider the following three scenarios: Fitness enthusiasts participating in exercise programs to build muscle mass, patients undergoing rehabilitation programs to regain their abilities, and pilots refining their flying skills to prepare for flight tasks. Each of these activities falls within the realm of the training industry, which is rapidly expanding due to the growing demand to improve physical well-being, professional expertise, and practical skills. For example, the global fitness training industry was valued at $160 billion in 2021, and it is projected to have 171.75% growth by 2028.[1] Rehabilitation training services are required by approximately 2.41 billion people worldwide, and the demand keeps growing.[2]

Usually, a training program involves multiple sessions, and each session contains multiple activities. For example, Mohan (2024), a skilled athlete, suggests that people should participate in the training program shown in Table 1 to maintain their muscle strength. There are three sessions in the program.

[1]Faculty of Business for Science & Technology, School of Management, University of Science and Technology of China, Hefei, China
[2]College of Business, University of Houston-Clear Lake, Houston, Texas, USA

Corresponding author:
Yifu Li, Faculty of Business for Science & Technology, School of Management, University of Science and Technology of China, Hefei 230026, China.
Email: yifuli@ustc.edu.cn

Table 1. Training program proposed by Mohan (2024).

No.	Session 1 Activity	Session 1 Repetitions	Session 2 Activity	Session 2 Repetitions	Session 3 Activity	Session 3 Repetitions
1	Squats	8 – 10	Bench press	10	Barbell row/deadlift	8 – 10
2	Leg extension	10	Incline bench press	10	Wide grip lat pulldown	10
3	Lying hamstring curl	10	Triceps pushdowns	10	Military press	10
4	Calf raises	10	Cardio	10 – 15 min	Barbell shrug	10
5	Barbell curl	10	Crunches	20	Cardio	10 – 15 min

Each session contains five activities, and each activity should be repeated multiple times in the recommended sequence. It is crucial for customers to participate in all of these activities to ensure their muscles are well-maintained.

However, many individuals fail to complete the training program, quitting after only a few sessions. In practice, 73% of fitness training participants give up before reaching their targets (Eason, 2013). For home-based rehabilitation training programs, the adherence rate is even lower—less than 40% (Peek et al., 2018). Participants abandon training programs for different reasons, but the most crucial one is that the participation decision is affected by the past training experience, which depends on both experiential and functional utilities. Experiential utility refers to the emotional experience of the customer. For instance, customers may feel fatigued or get a sense of accomplishment in a training program. Functional utility refers to the tangible physical benefits of the training activities. Both utilities affect the participation decision. For instance, customers may abandon the program if they feel over-fatigued or believe the training benefits are too low. In most situations, once a customer quits the training process, it becomes challenging for them to rejoin the program.[3] The abandonment behavior results in significant losses for both the customer and the service provider. For one thing, a customer cannot enjoy the benefits of training once they abandon the program. For another, the service provider's revenue will be diminished if a customer quits.

Operations management researchers have acknowledged the importance of training program design, leading to multiple studies on various training programs. For instance, Zwols and Sierksma (2009) examine the optimal training schedule of decathletes considering interrelated effects on the overall sports performance. De Bruecker et al. (2018) investigate the aircraft technician training problem, discussing the optimal training schedule for the desired skill mix. Roels (2020) introduces a fitness-fatigue model to characterize athletic performance and reveals an optimal practice process to maximize an athlete's performance at the end of the program. Forrest et al. (2022) develop an Air Force pilot training recommendation system that automatically generates the training plan. These studies explore the design of training programs to maximize participants' overall performance without considering abandonment behavior. We refer to these training programs as the *performance-oriented* training programs.

However, in many circumstances, retention rather than performance is the most important concern for the service designers. We refer to these programs as the *retention-oriented* programs. For example, after completing performance-oriented rehabilitation training in the clinic, patients must participate in retention-oriented training programs at home to maintain the skills they have acquired. Considering that the patients may abandon the training when the experience is unpleasant, the designer seeks to enlarge the number of training sessions the patients participate in, which benefits their body health. The market size for such retention-oriented rehabilitation training services is estimated to be $159.88 billion in 2025 and is projected to grow to about $274.23 billion by 2034.[4] While the aforementioned studies shed light on the performance-oriented training design, there is limited guidance on the design of retention-oriented training programs, despite their important role in the training industry, which contributes a large share of the revenue.

Recognizing the significance of the retention-oriented training program and the gaps in its design theory leads to the central question in this study: How can a retention-oriented training program be designed to maximize customer retention, considering the effect of the past training experience? To address this question, we develop an optimization problem that aims to maximize customer retention and propose a solution method to solve it.

First, we introduce a retention model in the framework of Aflaki and Popescu (2014), Bellos and Kavadias (2021), and Li et al. (2023). Each training activity has four attributes: *Reward*, *difficulty*, *value*, and *duration*. *Reward* refers to the sense of achievement that individuals obtain from the training progress. For instance, participants in a weight-loss program might feel a sense of accomplishment as a reward when they finish a two-hour exercise. *Difficulty* refers to the degree of challenge. Difficult activities make customers feel more tired and stressed than easy ones. *Value* refers to the amount of functional benefit that customers receive from the training. For instance, a fitness program can provide functional benefits such as enhancing muscle strength. *Duration* refers to the expected time required to complete the activity.

According to the literature (e.g., Aflaki and Popescu, 2014; Long et al., 2020), customer retention depends on the customer's previous service experience, which can be modeled by the remembered utility—a weighted sum of the instantaneous utilities. Similar to Bellos and Kavadias (2021), we consider an instantaneous holistic utility, which is a linear function of instantaneous experiential and functional utilities of a training program. Similar to Li et al. (2023), the instantaneous experiential utility is a linear combination of the instantaneous utility and disutility, which depends on the reward and difficulty of the activities subject to the adaptation process. Because the instantaneous functional utility reflects the tangible physical benefit, it depends solely on the value of a training activity. The remembered holistic utility is a stock of the instantaneous holistic utility subject to the memory decay process as considered by Das Gupta et al. (2016). Like Aflaki and Popescu (2014), we adopt the goodwill retention model, where the customer with a higher remembered holistic utility is more likely to participate in the next training session. Finally, we formulate the retention-oriented training program design problem (RTDP), in which the service designer seeks to optimize the retention of all training sessions.

Second, we investigate the solution approach for RTDP. Our analysis shows that RTDP is a 0–1 constrained exponential sum problem (0–1 CESP), which we prove to be NP-hard. Because the service provider may need an optimal solution of RTDP in the training program design practice, we focus on developing an algorithm to find such a solution for any RTDP instance. Usually, operations researchers rely on branch and bound (B&B) algorithms to solve the 0–1 CESP (e.g., Icmeli and Erenguc, 1996; Vanhoucke et al., 2001; Zhao et al., 2020). In these studies, the objective functions are typically expressed as $\sum_{i=1}^{n} \sum_{j=1}^{n} e^{c_j \pi_{i,j}}$, where c_j is a monotonic coefficient in j. The B&B algorithm adopts this monotonic property to compute the bounds of the objective function directly. Unfortunately, there is no such monotonic property in our case. Therefore, we solve the 0–1 CESP through another approach—the geometric branch and bound (GB&B) algorithm—which does not require computing the bounds of the objective function directly (Scholz, 2012). Unlike the B&B algorithm, the GB&B algorithm divides the objective function into multiple "dimensions" and calculates their bounds by solving subproblems, 0–1 mixed-integer linear programs (MILPs) in our case, which are less complex. This makes the GB&B algorithm an ideal solution method for the 0–1 CESP in our study and for many other 0–1 CESPs that cannot be solved by the standard B&B algorithm due to the difficulty of directly computing the bounds of objective functions.

To examine the performance of our proposed GB&B algorithm, we conduct a thorough numerical study on its computational efficiency. We find that the GB&B algorithm solves RTDP more efficiently than commercial solvers, including BARON, SCIP, and GUROBI. Our testing shows that GB&B can solve RTDP with 600 activities and eight sessions within an hour.

To gain managerial insights, we investigate the optimal structural properties when the reward, difficulty, and functional value are proportional to each other. We find that if the reward is high, there are more U-shaped structures in each session, which is similar to the case in Das Gupta et al. (2016). In this case, a steep positive gradient toward the end of the U-shaped sequence can enhance the customer experience. If the difficulty is high, most instances exhibit an inverted U-shaped or decreasing subsequence pattern. The reason is that scheduling low-difficulty activities at the beginning and end of the session provides a warm-up and cool-down for customers. If the value of each activity is high, there are more increasing subsequences in each session. In this case, the memory decay effect dominates the other effects. Customers value recent utilities more than previous ones, resulting in increasing subsequences.

We also discuss several extended problems that the designer may face in practice. First, we investigate RTDP with breaks between sessions, and study the optimal value of RTDP as the break duration varies. Our analysis reveals a nonmonotonic relationship between break duration and the optimal value, characterized by an initial increase followed by a subsequent decrease. Second, we consider the case that the service provider cares about both retention and performance, which leads to a joint training retention-performance optimization problem (JTOP). Our findings reveal that there are more pulsed subsequences when the service designers care about performance, and more nonpulsed subsequences when they focus on retention.

Even though our paper falls within the area of training program design, the model and solution algorithms we propose can be extended and applied in a broad area within the service industry. For example, video game designers (e.g., Lei et al., 2023; Li et al., 2023) may extend our model and algorithm to investigate the optimal design that enhances the retention of a single-player game. School and online education platforms may apply and modify our algorithm to schedule courses to enhance students' long-term retention.

The rest of this article is organized as follows. In Section 2, we summarize the relevant literature. Section 3 presents the main model. Section 4 analyzes the computational complexity of RTDP and discusses the solution methods. Section 5 describes the framework of our proposed GB&B algorithm. Section 6 presents our computational studies and discusses the managerial insights. In Section 7, we investigate extended problems. We conclude this article in Section 8.

2 Related Work

Our work falls into the realm of training program design (e.g., De Bruecker et al., 2018; Ladany, 1975; Qi et al., 2004; Roels, 2020; Zwols and Sierksma, 2009). Usually, this stream of literature concerns performance optimization over a given training period. We discuss several closely related papers in this area. For example, Zwols and Sierksma (2009) investigate the training time allocation problem, which optimizes decathlete performance considering the related effects of different exercises. De Bruecker et al. (2018) introduce a three-stage mixed integer programming method to resolve the optimal aircraft training schedule and skill mix. Roels (2020) studies the training intensity sequencing problem to maximize the athlete's performance at the end of the program, introducing a fitness-fatigue model for the training performance and deriving an optimal sequence in different scenarios. Eliaz and Spiegler (2021) propose an optimal stochastic training strategy based on the Markov process to maximize muscle mass. Our work differs from these studies in two ways. First, we study the design of retention-oriented training programs instead of performance-oriented programs. As such, we care more about retaining participants than improving their performance. Second, while former studies assume that participants will never quit the training program, we consider the scenario that they may abandon the program partway, which better fits training program realities.

Our study also belongs to the area of service operations management. An important question within this domain is how to enhance service retention. Scholars have introduced multiple methods such as providing different content (e.g., Aflaki and Popescu, 2014; Lei et al., 2023, 2024), improving matchmaking policies (e.g., Chen et al., 2021), and adjusting the price (e.g., Liao and Chen, 2021; Roels, 2014). We highlight several of the most relevant studies. For example, Aflaki and Popescu (2014) introduce a retention optimization problem, in which the service provider can adjust the service quality dynamically. Bernstein et al. (2022) study how a firm providing repeated services should design the content. Unlike these studies, our work optimizes service retention by selecting and scheduling training activities. In our study, we model the effect of previous experience on customer retention. In the literature, service experience optimization is also a key research question (e.g., Chen et al., 2024; Das Gupta et al., 2016; Deshmane et al., 2023; Li et al., 2022, 2023). Das Gupta et al. (2016) and Li et al. (2022) investigate service experience optimization considering the effects of adaptation and memory decay. Li et al. (2023) extend the discussion to the experience of interactive services and video games, deriving optimal sequences under different reward schemes. While these studies solely discuss the experience of a service, our study concerns both the experiential and functional value of a service, similar to Bellos and Kavadias (2021). However, unlike Bellos and Kavadias (2021), we introduce a detailed model considering the psychological effects on both service experience and retention. Finally, several studies consider the satiation effect in the service consumption process. For instance, Baucells and Sarin (2007) and Baucells and Sarin (2010) capture the customer's satiation effect due to previous consumption and investigate the optimal consumption sequence. Based on these works, Baucells and Zhao (2020) establish a comprehensive framework that formally characterizes the satiation model. Unlike these studies, we consider the adaptation and memory decay effects instead of the satiation effect.

Our article also contributes to the literature on service scheduling and optimization (e.g., Dixon and Thompson, 2016; Hall and Liu, 2023; Zhu et al. 2024; Li and Qi, 2022). To name a few related studies, Dixon and Thompson (2016) investigate the service bundle design problem and propose an efficient simulated annealing algorithm. Li and Qi (2022) continue the investigation and propose an exact algorithm to compute the optimal bundle assignment when the schedule of the activities is already given. By our analysis, the RTDP in our study is a 0–1 CESP, which is NP-hard. Few works in the domain discuss the solution methods of such a problem because of its computational complexity, and almost all solution approaches are based on the B&B framework (e.g., Icmeli and Erenguc, 1996; Vanhoucke et al., 2001; Wang, 1995; Zhao et al., 2020). Typically, these algorithms rely on the monotonic property of the coefficients in computing the bound of the objective function. For example, for the capacity expansion problem with exponential demand growth, Wang (1995) develops a B&B algorithm that computes the bounds of the objective function by the monotonic property of the demand projection. Zhao et al. (2020) propose a B&B algorithm for the project scheduling problem with exponential failure rates. It computes the bounds based on the monotonic property of the project failure rate. In our study, no special structure property can be applied to compute the bound of the objective function directly. Given this circumstance, we resolve RTDP with a GB&B algorithm, which does not require directly computing the bound of the objective function. This makes the GB&B algorithm an efficient solution method for the problem with a complicated objective function, such as the sum-of-ratio problem (e.g., Li and Qi, 2022; Ursulenko et al., 2013), and the sum of Euclidean distance problem (Schöbel and Scholz, 2014). Readers are referred to Scholz (2012) for a detailed review of applications of the algorithm. Our proposed algorithm proves to be efficient by the numerical studies, and it reveals the potential of the GB&B algorithm in solving 0–1 CESPs that cannot be resolved by the B&B algorithm because of the difficulty in computing the bounds.

3 Model

In this section, we introduce the model of the retention-oriented training design problem. The service provider selects and schedules n activities from a set of $N (n \leq N)$ given activities to optimize customer retention. We introduce a binary variable $\boldsymbol{\pi} = (\pi_{i,j})_{N \times n}$ to denote the service designer's selection and scheduling decisions. If activity i is selected and scheduled to the jth position, we have $\pi_{i,j} = 1$; otherwise, we have $\pi_{i,j} = 0$.

Each training activity $i \in \{1, \ldots, N\}$ has four associated attributes: reward level (reward) r_i, difficulty level (difficulty) d_i, function value (value) v_i, and duration τ_i. Reward and difficulty are commonly considered in the service design literature (e.g., Das Gupta et al., 2016; Li et al., 2022, 2023). The reward r_i represents the sense of achievement or pleasure a customer obtains from participating training activity i. The difficulty d_i reflects the amount of effort or energy a customer expends to complete training activity i. The value v_i is a well-accepted attribute (e.g., Bellos and Kavadias, 2021) representing the tangible benefits that training activity i delivers to customers, such as weight reduction in a weight-loss training program.

The duration τ_i is the expected time required to complete a training activity i. Following the literature (e.g., Aflaki and Popescu, 2014; Lei et al., 2023; Roels, 2020), we consider the scenario where each activity has an identical duration τ. This assumption fits the practice of many training programs. For instance, the fitness application *7 Minute Workout*, one of the best "Apple Apps for Health," offers training activities with an identical duration of seven minutes.[5] Another example comes from the *F45 Training*, in which all the fitness training classes last for 45 minutes.[6] Let T and t_j denote the duration of the whole training program and the completion time of the jth activity, respectively. We have $T = \tau n$ and $t_j = \tau j$ under the identical duration assumption. We let $t_0 = 0$ denote the start time of the training program. Finally, it is important to highlight that the values of these four attributes are all specified and fixed for the service designer.

We consider the case that the retention-oriented training program has $L > 1$ sessions,[7] with each session $l \in \{1, \ldots, L\}$ containing b_l activities, such that $\sum_{l=1}^{L} b_l = n$. It is worth noting that L plays an important role in the service design, which affects the optimal solution and the efficiency of the proposed algorithm. Let $B_l = \sum_{i=1}^{l} b_i$ denote the number of activities in the first l sessions and let $B_0 = 0$. We assume that the customer starts the training from the first training activity in the first session at t_0. During the retention-oriented training process, the customer may abandon the service after finishing a certain number of training sessions. The service designer makes the activity scheduling decision $\boldsymbol{\pi}$ before the commencement of the training to maximize overall customer retention.

We discuss the customer's utility and training retention in Sections 3.1 and 3.2, respectively. We formulate the service provider's decision problem in Section 3.3.

3.1 Customer Utility

In this section, we discuss the model of the customer utility considering the adaptation and memory decay effects. We illustrate the psychological process of customer utility in Figure 1.

The instantaneous holistic utility, which reflects the customer experience at time t, is a combination of the instantaneous functional and experiential utilities. The instantaneous functional utility solely depends on the value of the activities. Differently, the instantaneous experiential utility depends on the instantaneous utility and disutility, which are derived from the reward and difficulty of the activities subject to the adaptation process. Adaptation reflects a psychological phenomenon in which people become accustomed to states but react to changes. We discuss the model of the instantaneous holistic utility in Section 3.1.1.

The remembered holistic utility, depicting the customer's satisfaction with the training program, is an integration of instantaneous holistic utilities. It is affected by the memory decay process, which refers to the memory loss of customers. We discuss the model of remembered holistic utility in Section 3.1.2.

3.1.1 Instantaneous Holistic Utility.

The instantaneous holistic utility depends on both instantaneous functional and experiential utilities denoted by $v_f(t)$ and $v_e(t)$, respectively. They refer to the functional benefit and emotional value of the customer at time t. Similar to Bellos and Kavadias (2019), Bellos and Kavadias (2021), the instantaneous holistic utility $v(t)$ is a linear function of these two utilities, which is given by

$$v(t) \triangleq \delta_f v_f(t) + \delta_e v_e(t), \tag{1}$$

where δ_f and δ_e are the coefficients of instantaneous functional and experiential utilities, respectively.

We now discuss the formulation of instantaneous functional and experiential utilities. According to the literature (Das Gupta et al., 2016; Li et al., 2022, 2023), the instantaneous experiential utility is subject to the adaptation process, which makes participants seek an improved experience. However, the instantaneous functional utility is independent of the adaptation process because the target of retention-oriented training is not enhancing performance but maintaining the physical condition or skill level. Therefore, customers do not care whether they obtain more functional value than before. Hence, their functional utility is not subject to the adaptation process.

(i) Instantaneous Functional Utility

The instantaneous functional utility $v_f(t)$ at time $t \in (t_{i-1}, t_i]$ depends on the value of the training activity scheduled at time t, which is given by

$$v_f(t) = v_{(i)}, \tag{2}$$

where $v_{(i)}$ is the value of the ith activity in the training service.

$v_f(t)$ is a constant during $(t_{i-1}, t_i]$, for all $i \in \{1, \ldots, n\}$. This is consistent with Bellos and Kavadias (2021), where the instantaneous functional utility at time t depends on the value of the activity at time t.

(ii) Instantaneous Experiential Utility

Similar to Das Gupta et al. (2016) and Li et al. (2022, 2023), we consider the adaptation process on the instantaneous experiential utility. Given a constant reward (difficulty), a customer will gradually become accustomed to that reward (difficulty) level, so the instantaneous utility (disutility) will decrease over time and eventually become zero.

Let $f_r(t)$ and $f_d(t)$ denote the reference levels of reward and difficulty at time t, respectively. As in Das Gupta et al. (2016), the instantaneous utility $u_r(t)$ (disutility $u_d(t)$) at time t is the difference between the current reward (difficulty) and the reference reward (difficulty). The instantaneous utility $u_r(t)$ and

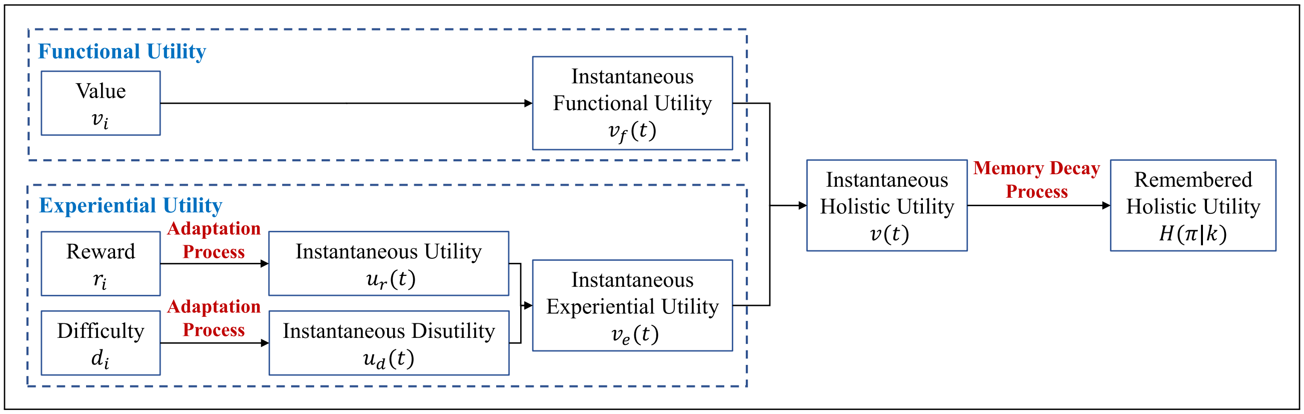

Figure 1. Psychological process in a retention-oriented training program.

disutility $u_d(t)$ at time $t \in [t_{i-1}, t_i]$ are given by

$$u_r(t) = r_{(i)} - f_r(t), \tag{3}$$
$$u_d(t) = d_{(i)} - f_d(t), \tag{4}$$

where $r_{(i)}$ and $d_{(i)}$ are the reward and difficulty of the ith activity in the training program, respectively.

Similar to Das Gupta et al. (2016) and Li et al. (2022, 2023), the change rate of the reference level is proportional to the difference between the instantaneous utility (disutility) and the reference level. Let $\alpha > 0$ and $\beta > 0$ denote the adaptation rates of reward and difficulty, respectively, indicating the speed of adaptation. The change rates of the reference levels at time $t \in [t_{i-1}, t_i]$ are given by

$$\frac{df_r(t)}{dt} = \alpha(r_{(i)} - f_r(t)), \tag{5}$$
$$\frac{df_d(t)}{dt} = \beta(d_{(i)} - f_d(t)). \tag{6}$$

Given (5) and (6), the instantaneous utility and disutility at $t \in [t_{i-1}, t_i]$ can be expressed as

$$u_r(t) = \left((r_{(1)} - f_r(0)) + \sum_{j=2}^{i} (r_{(j)} - r_{(j-1)}) e^{\alpha t_{j-1}} \right) e^{-\alpha t}, \tag{7}$$
$$u_d(t) = \left((d_{(1)} - f_d(0)) + \sum_{j=2}^{i} (d_{(j)} - d_{(j-1)}) e^{\beta t_{j-1}} \right) e^{-\beta t}, \tag{8}$$

where $f_r(0)$ and $f_d(0)$ are the initial reward and difficulty reference levels, respectively. In the base model, we assume customers are homogeneous with the same $f_r(0)$ and $f_d(0)$. In E-companion EC.4, we discuss the extension where customers are heterogeneous with different $f_r(0)$ and $f_d(0)$.

As in Li et al. (2023) and Roels (2020), the instantaneous experiential utility $v_e(t)$ at time t is a linear combination of $u_r(t)$ and $u_d(t)$, which is given by

$$v_e(t) = k_r u_r(t) - k_d u_d(t), \tag{9}$$

where $k_r > 0$ and $k_d > 0$ are coefficients of the instantaneous utility and instantaneous disutility. As in Li et al. (2023), we normalize k_r and k_d to 1 without loss of generality.

3.1.2 *Remembered Holistic Utility.*

The remembered holistic utility $H(\boldsymbol{\pi}|k)$ reflects customers' satisfaction with the training program at the end of the kth activity. Similar to Baucells and Zhao (2019) and Das Gupta et al. (2016), the remembered holistic utility is a stock of the instantaneous holistic utilities with memory decay. The memory decay process is a common psychological phenomenon, which depicts that customers gradually forget past experiences over time (Roels, 2020). With such an effect, customers remember the recent activities better than the previous ones. We adopt the exponential decay model proposed by Ebbinghaus (2013) and used by Das Gupta et al. (2016) and Roels (2020), where the customers discount past experiences exponentially. The remembered holistic utility $H(\boldsymbol{\pi}|k)$ is given by

$$H(\boldsymbol{\pi}|k) = \sum_{i=1}^{k} \int_{t_{i-1}}^{t_i} v(t) e^{-\omega(t_k - t)} dt,$$

where parameter $\omega \in [0, \infty)$ is the memory decay rate of the customer.

Combining (1), (2), and (7) to (9), the remembered holistic utility $H(\boldsymbol{\pi}|k)$ can be expressed by

$$\begin{aligned} H(\boldsymbol{\pi}|k) &= \delta_e \left(\sum_{i=1}^{N} \sum_{j=1}^{k} \pi_{i,j} r_i \left(\frac{e^{-\alpha(k-j+1)\tau} - e^{-\omega(k-j+1)\tau}}{\omega - \alpha} \right.\right. \\ &\quad \left.\left. - \frac{e^{-\alpha(k-j)\tau} - e^{-\omega(k-j)\tau}}{\omega - \alpha} \right) - f_r(0) \frac{e^{-\alpha k\tau} - e^{-\omega k\tau}}{\omega - \alpha} \right) \\ &\quad - \delta_e \left(\sum_{i=1}^{N} \sum_{j=1}^{k} \pi_{i,j} d_i \left(\frac{e^{-\beta(k-j+1)\tau} - e^{-\omega(k-j+1)\tau}}{\omega - \beta} \right.\right. \end{aligned}$$

$$-\frac{e^{-\beta(k-j)\tau}-e^{-\omega(k-j)\tau}}{\omega-\beta}\Bigg)-f_d(0)\frac{e^{-\alpha k\tau}-e^{-\omega k\tau}}{\omega-\beta}\Bigg)$$
$$+\delta_f\sum_{i=1}^{N}\sum_{j=1}^{k}\pi_{i,j}v_i\left(\frac{1-e^{-\omega(k-j+1)\tau}}{\omega}-\frac{1-e^{-\omega(k-j)\tau})}{\omega}\right).$$

For simplicity, we introduce functions $F_1(\boldsymbol{\pi}|k)$, $F_2(\boldsymbol{\pi}|k)$, and $F_3(\boldsymbol{\pi}|k)$, given by

$$F_1(\boldsymbol{\pi}|k)=\sum_{i=1}^{N}\sum_{j=1}^{k}\pi_{i,j}r_i\left(\frac{e^{-\alpha(k-j+1)\tau}-e^{-\omega(k-j+1)\tau}}{\omega-\alpha}-\frac{e^{-\alpha(k-j)\tau}-e^{-\omega(k-j)\tau})}{\omega-\alpha}\right)-f_r(0)\frac{e^{-\alpha k\tau}-e^{-\omega k\tau}}{\omega-\alpha}, \quad (10)$$

$$F_2(\boldsymbol{\pi}|k)=\sum_{i=1}^{N}\sum_{j=1}^{k}\pi_{i,j}d_i\left(\frac{e^{-\beta(k-j+1)\tau}-e^{-\omega(k-j+1)\tau}}{\omega-\beta}-\frac{e^{-\beta(k-j)\tau}-e^{-\omega(k-j)\tau}}{\omega-\beta}\right)-f_d(0)\frac{e^{-\alpha k\tau}-e^{-\omega k\tau}}{\omega-\beta}, \quad (11)$$

$$F_3(\boldsymbol{\pi}|k)=\sum_{i=1}^{N}\sum_{j=1}^{k}\pi_{i,j}v_i\left(\frac{1-e^{-\omega(k-j+1)\tau}}{\omega}-\frac{1-e^{-\omega(k-j)\tau})}{\omega}\right). \quad (12)$$

Here, $F_1(\boldsymbol{\pi}|k)$, $F_2(\boldsymbol{\pi}|k)$, and $F_3(\boldsymbol{\pi}|k)$ reflect the remembered accomplishment, fatigue, and functional level over the first k activities. With (10) to (12), $H(\boldsymbol{\pi}|k)$ can be rewritten as

$$H(\boldsymbol{\pi}|k)=\delta_e F_1(\boldsymbol{\pi}|k)-\delta_e F_2(\boldsymbol{\pi}|k)+\delta_f F_3(\boldsymbol{\pi}|k).$$

3.2 Customer Training Retention

Similar to Aflaki and Popescu (2014), customers make their participation decision in a backward-looking mode. The customers' decision at the end of session l depends on their holistic remembered utility of past sessions $H(\boldsymbol{\pi}|B_l)$ and the utility of their outside option U_o, which reflects the pleasure consumers obtain from participating in an alternative service. Similar to Aflaki and Popescu (2014), Long et al. (2020), and Liao and Chen (2021), the customer will participate in the next session (i.e., session $l+1$) if $H(\boldsymbol{\pi}|B_l)\ge U_o$.

Following Alptekinoğlu and Semple (2016) and Ben Rhouma and Zaccour (2018), we assume that U_o is independent of the session l and heterogeneous across the customers, which follows an exponential distribution with rate parameter $\lambda>0$, such that $\mathbb{P}(U_o\le u_o)=1-e^{-\lambda u_o}$. For simplicity, we introduce $H_l(\boldsymbol{\pi})$ to represent $H(\boldsymbol{\pi}|B_l)$. Since U_o is heterogeneous, with a given $H_l(\boldsymbol{\pi})$, certain customers may continue the training while others abandon the program. We denote by $R_l(\boldsymbol{\pi})$ the retention rate of the customer at the end of session l. Because a customer will participate in the next session only if $H_l(\boldsymbol{\pi})\ge U_o$, we have $R_l(\boldsymbol{\pi})=\mathbb{P}(H_l(\boldsymbol{\pi})\ge U_o)=\mathbb{P}(U_o\le H_l(\boldsymbol{\pi}))=1-e^{-\lambda H_l(\boldsymbol{\pi})}$. As $\lambda>0$, the retention rate $R_l(\boldsymbol{\pi})$ is increasing in $H_l(\boldsymbol{\pi})$. This is consistent with the goodwill retention model discussed in Aflaki and Popescu (2014). It reflects the fact that customers with a better holistic experience are more likely to participate in the next session, which is commonly considered in the literature (e.g., Kanoria et al., 2024).

Finally, let $\hat{R}_l(\boldsymbol{\pi})$ be the retention rate by the end of the session $l\in\{1,\ldots,L\}$:

$$\hat{R}_l(\boldsymbol{\pi})=\prod_{\mathrm{i}=1}^{l}R_i(\boldsymbol{\pi}).$$

3.3 Provider's Training Program Design Problem

In this section, we discuss the service provider's decision problem. The service provider has different weights for the retention rate in different training sessions. Let $w_l\ge 0$ denote the weight of the retention rate by the end of the session l. The weighted retention rate is given by

$$\Pi(\boldsymbol{\pi})=\sum_{l=1}^{L}w_l\hat{R}_l(\boldsymbol{\pi})=\sum_{l=1}^{L}w_l\prod_{\mathrm{i}=1}^{l}(1-e^{-\lambda H_i(\boldsymbol{\pi})}).$$

The service provider decides on the selection and schedule of training activities to maximize the weighted retention. The RTDP is given by

$$\max_{\boldsymbol{\pi}}\ \Pi(\boldsymbol{\pi})=\sum_{l=1}^{L}w_l\prod_{i=1}^{l}(1-e^{-\lambda H_i(\boldsymbol{\pi})}), \quad (13)$$
$$\text{s.t. } H_l(\boldsymbol{\pi})=\delta_e F_1(\boldsymbol{\pi}|B_l)-\delta_e F_2(\boldsymbol{\pi}|B_l)+\delta_f F_3(\boldsymbol{\pi}|B_l), \quad (14)$$
$$\sum_{i=1}^{N}\pi_{i,j}=1,\quad \forall j\in\{1,\ldots,n\}, \quad (15)$$
$$\pi_{i,j}\in\{0,1\},\quad \forall i\in\{1,\ldots,N\},\ \forall j\in\{1,\ldots,n\}, \quad (16)$$
$$\boldsymbol{\pi}\in\mathcal{P}. \quad (17)$$

We can observe that RTDP is a 0–1 CESP with the summation of exponential terms in the objective function. In RTDP, constraint (14) represents the remembered holistic utility $H_l(\boldsymbol{\pi})$. Constraint (15) guarantees that exactly one activity needs to be assigned to each position. Constraint (16) restricts the decision variable $\pi_{i,j}$ to be binary. In practice, the service provider may consider several practical constraints. Constraint (17) ensures the feasible schedule should satisfy these practical constraints, where $\mathcal{P}$ is a polytope restricted by them. For example, the service designer may consider the maximum repeated assignment and minimum sum value constraints as follows.

$$\sum_{j=1}^{n}\pi_{i,j}\le\psi,\quad \forall i\in\{1,\ldots,N\}, \quad (18)$$

$$\sum_{i=1}^{N} \sum_{j=B_{l-1}+1}^{B_l} v_i \pi_{i,j} \geq \underline{v}, \quad \forall l \in \{1, \ldots, L\}, \tag{19}$$

where $\psi \geq 1$ is the maximum number of times each activity can be scheduled and $\underline{v} \geq 0$ is the minimum value required by each training session. Constraint (18) guarantees that each activity is selected less than ψ times. Constraint (19) requires the total value of activities scheduled in each session exceeding the minimum functional value $\underline{v}$. In E-companion EC.3.1, we conduct a sensitivity analysis on parameters ψ and $\underline{v}$. One thing worth mentioning is that the service provider may drop all these practical constraints and resolve an RTDP with only constraints (14) to (16) in certain circumstances.

4 Computational Complexity and Solution Methods

In this section, we discuss the computational complexity of RTDP and its solution methods. We analyze the computational complexity in Theorem 1.

THEOREM 1. *RTDP is NP-hard.*

Theorem 1 states that RTDP belongs to the class of NP-hard problems. The reason is that RTDP is a 0–1 CESP that can be reduced to the unconstrained binary polynomial problem (UBP), as we show in E-companion EC.1. The UBP has been proven to be an NP-hard problem by Elloumi et al. (2021).

RTDP is a nonlinear optimization problem, which is challenging to resolve. Some previous works (e.g., Icmeli and Erenguc, 1996; Vanhoucke et al., 2001; Zhao et al., 2020) adopt the B&B algorithm to solve the 0–1 CESP, which requires an efficient bounding technique for the objective function. In nearly all the proposed algorithms, the objective function of their decision problems can be expressed as $\sum_{i=1}^{N} \sum_{j=1}^{n} e^{c_j \pi_{i,j}}$, where $\pi_{i,j} \in \{0, 1\}$ is the decision variable, and c_j is a coefficient monotonic in j. The bounds of the objective function can be computed based on this monotonic property. We take Zhao et al. (2020) as an example. First, exponential terms in the objective function are transferred into nonlinear constraints with additional variables. These nonlinear constraints are converted into linear ones based on the property that $c_j \geq c_{j-1}$ for all $j \in \{2, \ldots, n\}$. This converts the problem into an MILP, whose bounds can be calculated efficiently. However, there is no such monotonic structure in RTDP, which prevents us from computing the bounds of the objective function directly. Therefore, we do not discuss the application of the B&B algorithm; rather, we seek other methods to solve RTDP.

We list the other solution approaches as follows:

(1) Mixed integer nonlinear programming (MINLP) solvers
Since the 0–1 CESP is a MINLP, we can use an MINLP solver to solve RTDP. Among these solvers, BARON and SCIP are recognized as the fastest and most robust (Bestuzheva et al., 2025; Mittelmann, 2021).

(2) MILP solvers
Certain MILP solvers, such as GUROBI, can solve the 0–1 CESP. GUROBI solves it by applying the piecewise-linear approximation method, which converts the 0–1 CESP into a series of MILPs by computing the linear bounds of the nonlinear terms in the 0–1 CESP.[8]

(3) The GB&B algorithm
The GB&B algorithm is a modification of the B&B algorithm. It decomposes the objective function into several dimensions. Each dimension is a part of the objective function. In each iteration, instead of directly calculating the bounds of the whole objective function, the algorithm computes the single-dimension optimization problem within a given box area. Based on the optimal value of the single-dimension optimization problem, it cuts the box into a smaller one or partitions it into two sub-boxes. The algorithm updates the bounds of the objective value based on the optimal value of the single-dimension optimization problems. Given that the GB&B algorithm does not require computing the bounds of the objective function directly, it is suitable for solving problems with a complicated objective function (e.g., Li and Qi, 2022; Ursulenko et al., 2013), such as RTDP in our case.

Based on the numerical studies in Section 6, the GB&B algorithm performs most efficiently among the aforementioned methods. Therefore, we focus on discussing the development of the GB&B algorithm.

5 GB&B Algorithm

In this section, we discuss the framework of our GB&B algorithm. We start by introducing the preliminaries in Section 5.1. We discuss the configuration of the GB&B algorithm in Section 5.2. Finally, we present the framework of our algorithm in Section 5.3.

5.1 Preliminaries

For convenience of analysis, we first rewrite the objective function of RTDP. Let Ω denote the power set of the set $\{H_1(\boldsymbol{\pi}), H_2(\boldsymbol{\pi}), \ldots, H_L(\boldsymbol{\pi})\}$ excluding the empty subset. The number of elements in set Ω is

$$|\Omega| = \sum_{i \in \{1,\ldots,L\}} \binom{L}{i} = 2^L - 1.$$

The exponent of each exponential term in (13) is a linear combination of elements in the corresponding subset of Ω. Therefore, there are also $|\Omega| = 2^L - 1$ exponential terms in the objective function of RTDP. Thus, we can reformulate (13) as

$$\Pi(\boldsymbol{\pi}) = \sum_{l\in\{1,\ldots,L\}} w_l + \sum_{s\subseteq\Omega}(-1)^{|s|}e^{-\lambda\sum_{i\in\{j|H_j(\boldsymbol{\pi})\in s\}}H_i(\boldsymbol{\pi})} \times \sum_{i\in\{\max\{j|H_j(\boldsymbol{\pi})\in s\},\ldots,L\}} w_i. \tag{20}$$

Define $\mathcal{D} = \{1, \ldots, 2^L - 1\}$. We introduce a function $g_k(\cdot)$ for $k \in \{0\} \cup \mathcal{D}$, which is given by

$$g_k(\boldsymbol{\pi}) = \begin{cases} \sum_{l\in\{1,\ldots,L\}} w_l, & \text{if } k = 0, \\ (-1)^{|s_k|}e^{-\lambda\sum_{i\in\{j|H_j(\boldsymbol{\pi})\in s_k\}}H_i(\boldsymbol{\pi})} \times \sum_{i\in\{\max\{j|H_j(\boldsymbol{\pi})\in s_k\},\ldots,L\}} w_i, & \text{if } k \in \mathcal{D}, \end{cases} \tag{21}$$

where s_k is the kth element in Ω. We can tell that $g_0(\boldsymbol{\pi})$ is a given constant because the weight parameter w_l is given.

With (21), function (20) can be simplified as

$$\Pi(\boldsymbol{\pi}) = g_0 + \sum_{k\in\mathcal{D}} g_k(\boldsymbol{\pi}).$$

Therefore, RTDP can be rewritten as

$$\max_{\boldsymbol{\pi}} \Pi(\boldsymbol{\pi}) = g_0 + \sum_{k\in\mathcal{D}} g_k(\boldsymbol{\pi}), \tag{22}$$

$$\text{s.t. } \boldsymbol{\pi} \in \mathcal{T}, \tag{23}$$

where $\mathcal{T}$ is the feasible set restricted by the linear constraints (14) to (17).

We refer to each exponential function $g_k(\boldsymbol{\pi})$ as the dimension $k \in \mathcal{D}$ of the objective function. There are $2^L - 1$ dimensions in the objective function. Instead of computing the bound of the objective function (22) directly, the GB&B algorithm solves the problem by dividing the feasible region into multiple "box" areas. In each iteration, it computes the optimal value of a selected dimension within a selected box. Based on this optimal value, it cuts the box into a smaller box or splits it into two sub-boxes. The algorithm repeats the computation and box partition in the next iteration given another dimension and box. We define the box in the mth iteration as $V_m = \{\boldsymbol{v} \in \mathbb{R}^D : L_k(V_m) \le v_k \le U_k(V_m), \forall k \in \mathcal{D}\}$, where $L_k(V_m)$ and $U_k(V_m)$ represent the lower and upper bounds of the dimension k over box V_m, respectively. RTDP over box V_m is given by

$$\max_{\boldsymbol{\pi}} \Pi(\boldsymbol{\pi}|V_m) = g_0 + \sum_{k\in\mathcal{D}} g_k(\boldsymbol{\pi}), \tag{24}$$

$$\text{s.t. } g_k(\boldsymbol{\pi}) \ge L_k(V_m), \quad \forall k \in \mathcal{D}, \tag{25}$$

$$g_k(\boldsymbol{\pi}) \le U_k(V_m), \quad \forall k \in \mathcal{D}, \tag{26}$$

$$\boldsymbol{\pi} \in \mathcal{T}. \tag{27}$$

However, solving problem (24) to (27) directly is still challenging. Therefore, the GB&B algorithm computes the upper bound of $\Pi(\boldsymbol{\pi}|V_m)$ instead. Let $\zeta(V_m)$ denote the upper bound, which is given by

$$\zeta(V_m) = g_0 + \sum_{k\in\mathcal{D}} \overline{g}_{k,V_m}, \tag{28}$$

where $\overline{g}_{k,V_m}$ is the upper bound of $g_k(\boldsymbol{\pi})$ over box V_m.

To compute $\overline{g}_{k,V_m}$ on dimension k over box V_m, similar to Scholz (2012) and Li and Qi (2022), we solve a subproblem given by

$$\max_{\boldsymbol{\pi}} g_k(\boldsymbol{\pi}), \tag{29}$$

$$\text{s.t. } g_i(\boldsymbol{\pi}) \ge L'_i(V_m), \quad \forall i \in \mathcal{D}, \tag{30}$$

$$g_i(\boldsymbol{\pi}) \le U'_i(V_m), \quad \forall i \in \mathcal{D}, \tag{31}$$

$$\boldsymbol{\pi} \in \mathcal{T}, \tag{32}$$

where $L'_k(V_m) = \max\{L_k(V_m), 0\}$ and $U'_k(V_m) = \max\{U_k(V_m), 0\}$ if $|s_k|$ is even, while $L'_k(V_m) = \min\{L_k(V_m), 0\}$ and $U'_k(V_m) = \min\{U_k(V_m), 0\}$ if $|s_k|$ is odd. We consider constraints (30) and (31) in the aforementioned subproblem because constraints (25) and (26) can be tightened based on the value of $|s_k|$. By (21), if $|s_k|$ is even, we have $g_k(\boldsymbol{\pi}) \ge 0$. There must be $\max\{L_k(V_m), 0\} = L'_k(V_m) \le g_k(\boldsymbol{\pi}) \le U'_k(V_m) = \max\{U_k(V_m), 0\}$. If $|s_k|$ is odd, we have $g_k(\boldsymbol{\pi}) \le 0$. In this case, we have $\min\{L_k(V_m), 0\} = L'_k(V_m) \le g_k(\boldsymbol{\pi}) \le U'_k(V_m) = \min\{U_k(V_m), 0\}$.

Suppose that $\boldsymbol{\pi}^*_{k,V_m}$ is the optimal solution of problem (29) to (32). Then we have $\overline{g}_{k,V_m} = g_k(\boldsymbol{\pi}^*_{k,V_m})$. Considering all $\overline{g}_{k,V_m}$ for all $k \in D$, we can compute the upper bound $\zeta(V_m)$ by (28). Since all feasible boxes collectively cover the feasible region of RTDP, the upper bound of the optimal value of RTDP is the maximum of $\zeta(V_m)$ among all feasible boxes. Because $\boldsymbol{\pi}^*_{k,V_m}$ is a feasible solution for the RTDP, the lower bound of the optimal value of RTDP is the maximum of $\Pi(\boldsymbol{\pi}^*_{k,V_m})$ for all dimensions k and boxes V_m. The GB&B algorithm repeats the computation until the upper and lower bounds of the objective value meet or the list of boxes is empty.

5.2 Configuration of the Algorithm

In this section, we describe the configuration of our GB&B algorithm.

5.2.1 Initialization. In this stage, the algorithm computes the bound of the initial box V_0, and it saves V_0 in the list $\mathcal{V}$. Then, the algorithm computes the lower bound $\underline{\Pi}$ and upper bound $\overline{\Pi}$ of RTDP.

5.2.2 Dimension and Box Selection. At the beginning of each iteration, the algorithm selects a dimension and a box to search for the optimal solution. For the dimension selection, the two methods usually adopted by the researchers are rotation-dimension search (RDS) and fixed-dimension search (FDS).

RDS selects the different dimensions one by one in each iteration (e.g. Schöbel and Scholz, 2014; Scholz, 2012), while FDS selects the same dimension in all iterations (Li and Qi, 2022). Numerical testing shows that RDS is more efficient in solving RTDP, so we adopt RDS to conduct the dimension selection.

For our box selection, the algorithm selects the box with the largest $\zeta(V_m)$ to continue the search, a technique commonly adopted by the literature (e.g., Li and Qi, 2022; Schöbel and Scholz, 2014).

5.2.3 Bounding. In this operation, the algorithm updates the upper bound of the selected dimension over the box by solving the corresponding subproblem. It also updates $\overline{\Pi}$ with the maximum value of $\zeta(V_m)$ for all V in list $\mathcal{V}$ and $\underline{\Pi}$ if the objective value of the incumbent solution is better than $\underline{\Pi}$.

In the mth iteration, given dimension k and box V_m, the algorithm should solve the single-dimension optimization problem (29) to (32) to compute the upper bound $\overline{g}_{k,V_m}$. However, since the exponential function is monotonically increasing, instead of solving problem (29) to (32) directly, we can simplify the computation by solving an equivalent problem. We now formulate this equivalent problem.

We introduce a function $f_k(\boldsymbol{\pi})$ for $k \in \{0\} \cup \mathcal{D}$, defined as

$$f_k(\boldsymbol{\pi}) = \begin{cases} \sum_{l \in \{1,\dots,L\}} w_l, & \text{if } k = 0, \\ (-1)^{|s_k|+1} \lambda \sum_{i \in \{j | H_j(\boldsymbol{\pi}) \in s_k\}} H_i(\boldsymbol{\pi}), & \text{if } k \in \mathcal{D}. \end{cases} \tag{33}$$

Given that $H_i(\boldsymbol{\pi})$ is a linear function of $\boldsymbol{\pi}$, $f_k(\boldsymbol{\pi})$ is also a linear function in $\boldsymbol{\pi}$. Let $\widehat{L}_k(V_m)$ and $\widehat{U}_k(V_m)$ denote the lower and upper bounds of $f_k(\boldsymbol{\pi})$ over box V_m, respectively. We have

$$\widehat{L}_k(V_m) = \begin{cases} \sum_{l \in \{1,\dots,L\}} w_l, & \text{if } k = 0, \\ (-1)^{|s_k|} \left(\ln |L'_k(V_m)| - \ln \left(\sum_{i \in \{\max\{j | H_j(\boldsymbol{\pi}) \in s_k\},\dots,L\}} w_i \right) \right), & \text{if } k \in \mathcal{D}, \end{cases} \tag{34}$$

$$\widehat{U}_k(V_m) = \begin{cases} \sum_{l \in \{1,\dots,L\}} w_l, & \text{if } k = 0, \\ (-1)^{|s_k|} \left(\ln |U'_k(V_m)| - \ln \left(\sum_{i \in \{\max\{j | H_j(\boldsymbol{\pi}) \in s_k\},\dots,L\}} w_i \right) \right), & \text{if } k \in \mathcal{D}. \end{cases} \tag{35}$$

We then introduce the optimization problem RTDP-(k,V_m), which is given by

$$\max_{\boldsymbol{\pi}} f_k(\boldsymbol{\pi}), \tag{36}$$

$$\text{s.t. } f_i(\boldsymbol{\pi}) \geq \widehat{L}_i(V_m), \quad \forall i \in \mathcal{D}, \tag{37}$$

$$f_i(\boldsymbol{\pi}) \leq \widehat{U}_i(V_m), \quad \forall i \in \mathcal{D}, \tag{38}$$

$$\boldsymbol{\pi} \in \mathcal{T}. \tag{39}$$

We prove that problem (29) to (32) and RTDP-(k, V_m) are equivalent, as stated in Lemma 1.

LEMMA 1. *An optimal solution for RTDP-(k, V_m) is optimal for problem* (29) *to* (32)*, and vice versa.*

Based on Lemma 1, we can compute the optimal solution of (29) to (32), and the upper bound $\overline{g}_{k,V_m}$ by solving RTDP-(k, V_m). Since $f_k(\boldsymbol{\pi})$ is linear, it is more efficient to solve RTDP-(k, V_m) than problem (29) to (32).

5.2.4 Branching. Our GB&B algorithm conducts the branching by dividing the box. According to Scholz (2012), when there are multiple dimensions, it is more effective to split the box into two sub-boxes. This suits our scenario, where the number of dimensions $|\mathcal{D}| = 2^L - 1$. Therefore, we conduct the branching by dividing the box into up to two sub-boxes.

We follow the splitting rule in Ursulenko et al. (2013) and Li and Qi (2022) for the branching. If the optimal solution of the subproblem computed by the bounding operation is located in the interior of the box, it will be cut into a smaller sub-box. Otherwise, if this optimal solution is located at the boundary of the box, it will be divided into two smaller sub-boxes.

5.2.5 Pruning. At the end of each iteration, boxes that cannot contain the optimal solution are removed from the list $\mathcal{V}$.

5.3 Framework of the Algorithm

In this section, we discuss the framework of our GB&B algorithm, which is presented in Algorithm 1. Figure 2 illustrates the general idea of the algorithm.

Steps 1 to 6 are the initialization process. The GB&B algorithm computes the upper bound of each dimension $k \in \mathcal{D}$ and assigns its optimal values as the upper bounds of the initial box V_0. Then, it adds the initial box V_0 to the box list $\mathcal{V}$. The algorithm sets the upper bound of the objective $\overline{\Pi}$ as $\zeta(V_0)$ and sets the lower bound of the objective $\underline{\Pi}$ as the maximum objective value of the optimal solution of RTDP-(k,V_0) for all $k \in \mathcal{D}$.

Steps 7 to 29 are the main iterations. The algorithm selects the dimension k rotationally, and the box with the largest $\zeta(V)$ is selected to continue the search. That is, in the mth iteration, the algorithm selects box $V_m = \arg\max\{\zeta(V) | V \in \mathcal{V}\}$. After the selection, the algorithm deletes the chosen box V_m from list $\mathcal{V}$ and updates the lower bound $\boldsymbol{L}(V_m)$ of the box V_m. Then $\widehat{\boldsymbol{L}}(V_m)$ and $\widehat{\boldsymbol{U}}(V_m)$ with (34) and (35) are computed. In

Algorithm 1: The GB&B Algorithm.

Input:
Parameters n, N, L, α, β, ω, τ, r_i, d_i, v_i, for all $i \in \{1, \dots, n\}$, and the target gap ϵ.
Output:
The optimal schedule $\underline{\boldsymbol{\pi}}$ and the optimal value $\underline{\Pi}$.

1: Let V_0 be the initial box and $m = 0$; `// Initialization`
2: Set $\widehat{\boldsymbol{U}}(V_0) = \rho$ and $\widehat{\boldsymbol{L}}(V_0) = -\rho$, where ρ is a large positive number;
3: Solve RTDP-(k,V_0) for all $k \in \mathcal{D}$, obtain the optimal solution $\boldsymbol{\pi}^*_{k,V_0}$;
4: Set $U_k(V_0) = g_k(\boldsymbol{\pi}^*_{k,V_0})$ for all $k \in \mathcal{D}$;
5: Let $\mathcal{V}$ be a list of boxes and add V_0 to $\mathcal{V}$;
6: Set $\underline{\boldsymbol{\pi}} = \arg\max\{\Pi(\boldsymbol{\pi}^*_{k,V_0}) | k \in \mathcal{D}\}$, $\underline{\Pi} = \Pi(\underline{\boldsymbol{\pi}})$, and $\overline{\Pi} = \zeta(V_0)$;
7: **while** $\overline{\Pi} - \underline{\Pi} > \epsilon\overline{\Pi}$ and $\mathcal{V} \neq \emptyset$ **do**
8:   **for** $k \in \mathcal{D}$ **do** ; `// Dimension Selection`
9:   Set $m = m + 1$;
10:   Select box $V_m = \arg\max\{\zeta(V) | V \in \mathcal{V}\}$ and remove it from $\mathcal{V}$; ; `// Box Selection`
11:   Set $\boldsymbol{L}(V_m) = \boldsymbol{U}(V_m) + \underline{\Pi} - \overline{\Pi}$;
12:   Compute $\widehat{\boldsymbol{L}}(V_m)$ and $\widehat{\boldsymbol{U}}(V_m)$ based on $\boldsymbol{L}(V_m)$ and $\boldsymbol{U}(V_m)$;
13:   Solve RTDP-(k,V_m) and obtain the optimal solution $\boldsymbol{\pi}^*_{k,V_m}$; `// Bounding`
14:   **if** $\Pi(\boldsymbol{\pi}^*_{k,V_m}) > \underline{\Pi}$ **then**
15:     Set $\underline{\boldsymbol{\pi}} = \boldsymbol{\pi}^*_{k,V_m}$ and $\underline{\Pi} = \Pi(\boldsymbol{\pi}^*_{k,V_m})$; `// Update the Optimal Solution`
16:   **end if**
17:   **if** $U_k(V_m) > g_k(\boldsymbol{\pi}^*_{k,V_m})$ **then**
18:     Introduce a new box $V' = V_m$; `// Branching Case I`
19:     Update $U_k(V') = g_k(\boldsymbol{\pi}^*_{k,V_m})$ and add V' to list $\mathcal{V}$;
20:   **else**
21:     Introduce two new boxes V' and V'', where $V' = V_m$ and $V'' = V_m$; `// Branching Case II`
22:     Let $j = \arg\max\{U_i(V_m) - g_i(\boldsymbol{\pi}^*_{k,V_m}) | i \in \mathcal{D} \text{ and } i \neq k\}$;
23:     Update $L_j(V') = (U_j(V_m) - g_j(\boldsymbol{\pi}^*_{k,V_m}))/2$;
24:     Update $U_j(V'') = (U_j(V_m) - g_j(\boldsymbol{\pi}^*_{k,V_m}))/2$;
25:     Add boxes V' and V'' to list $\mathcal{V}$;
26:   **end if**
27:   Remove boxes $\{V \in \mathcal{V} | \zeta(V) < \underline{\Pi}\}$ from list $\mathcal{V}$ and set $\overline{\Pi} = \max_{V \in \mathcal{V}} \zeta(V)$; `// Pruning`
28:   **end for**
29: **end while**
30: **return** $\underline{\boldsymbol{\pi}}$ and $\underline{\Pi}$.

the bounding process (step 13), the algorithm solves RTDP-(k,V_m). If the solution provides a larger objective value than $\underline{\Pi}$, the algorithm updates $\underline{\Pi}$ in step 15. The algorithm conducts the branching in steps 17 to 26 dividing the box into one or two smaller sub-boxes depending on the location of the solution in box V_m. If the optimal solution of the subproblem is found in the interior of the box V_m (i.e., when $U_k(V_m) > g_k(\boldsymbol{\pi}^*_{k,V_m})$), the current box is cut into a smaller box in step 18 to 19. If the optimal solution is located at the boundary of the box V_m (i.e., when $U_k(V_m) = g_k(\boldsymbol{\pi}^*_{k,V_m})$), the box V_m is split into two sub-boxes along the dimension with the longest distance between the incumbent solution and the upper bound of the dimension in steps 21 to 24. In this case, both V' and V'' are first initialized as V_m. Then the lower bound of dimension j in box V' and the upper bound of dimension j in box V'' are updated, where $j = \arg\max\{U_i(V_m) - g_i(\boldsymbol{\pi}^*_{k,V_m}) | i \in \mathcal{D} \text{ and } i \neq k\}$. In the pruning process (step 27), all boxes that cannot provide the optimal solution are removed from the box list. The algorithm updates the upper bound of the optimal value based on the latest box list $\mathcal{V}$.

The algorithm stops when the upper and lower bounds of the objective value meet (i.e., $\overline{\Pi} - \underline{\Pi} \leq \epsilon\overline{\Pi}$) or list $\mathcal{V}$ is empty, which indicates that the whole feasible region has been searched.

We prove the optimality and convergence of this GB&B algorithm in Theorem 2, which indicates that the algorithm finds an optimal solution after a finite number of iterations.

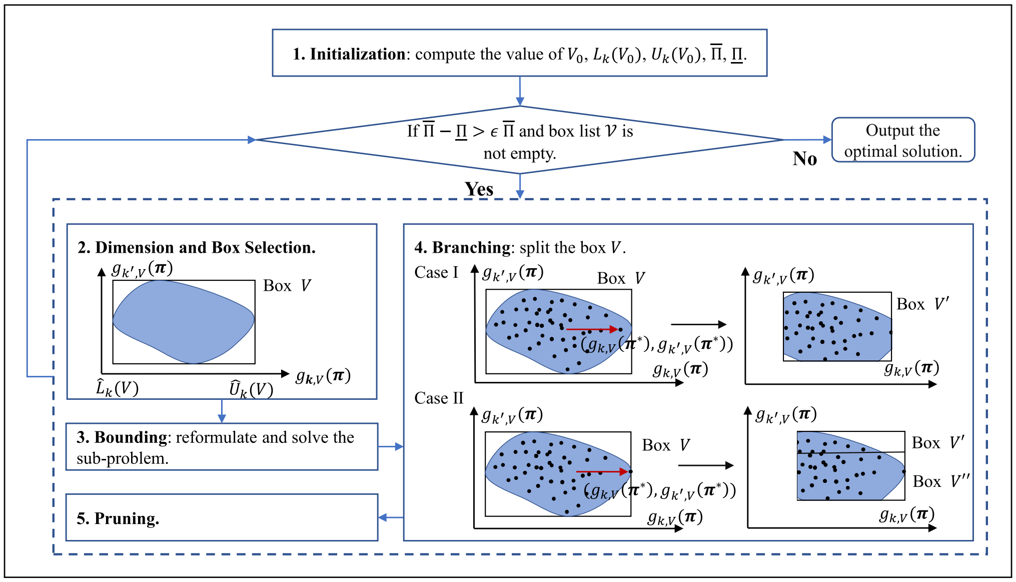

Figure 2. General idea of geometric branch and bound (GB&B) algorithm.

THEOREM 2. *This GB&B algorithm finds the optimal solution within a finite number of iterations.*

6 Computational Studies

In this section, we conduct several computational studies on the RTDP. We introduce the experiment setups in Section 6.1, and examine the performance of the GB&B algorithm in Section 6.2. In Section 6.3, we study the structural properties of the optimal sequence in the proportional scenario. In addition to the numerical studies discussed in this section, we present several other computational studies in E-companion EC.3.2 to EC.3.6, including RTDP with variable psychological coefficients, initial reference levels, activity duration, session weights, and attributes (i.e., reward, difficulty, and value).

6.1 Experiment Setups

In our experiments, following Li et al. (2023), we randomly generate reward r_i, difficulty d_i, and value v_i from the uniform distribution Uniform(0,1) for each training activity $i \in \{1, \dots, N\}$. Following Das Gupta et al. (2016) and Li et al. (2023), parameters α, β, and ω are sampled independently from the Gamma distribution Gamma(k_G, θ_G) with shape parameter $k_G = 2$ and scale parameter $\theta_G = 0.25$. We generate λ from a discrete set $\{0.2, 0.4, 0.6, 0.8, 1\}$ randomly, similar to Aflaki and Popescu (2014). Since we focus on activity scheduling, we conduct the numerical study where $n = N$ and each activity should be scheduled exactly once. There is no practical constraint (17) in these experiments. We set the activity's duration as $\tau = 1$ and coefficients of instantaneous experiential and functional utilities as $\delta_e = \delta_f = 1$. We configure the initial reference reward and difficulty level as $f_r(0) = f_d(0) = 0$. The weight parameter w_l is drawn randomly from the uniform distribution Uniform(0,1). We assume that each session contains an identical number of activities. For example, when $(n, L) = (30, 3)$, there are 10 activities in each session.

The computation time for each instance is limited to one hour, and we mark the runtime as "-" when it exceeds the time limit. We configure the tolerance of the optimal solution ϵ as 0.01% and generate 20 instances for each case.

We conduct all computational studies in JAVA 1.8.0, running on a computer with an Intel Core i5-14600 3.50GHz CPU and 64GB of RAM. RTDP-(k,V) is resolved by MILP solver GUROBI 11.0.0.

In Section 6.2, we conduct the experiment with BARON 23.6.23 and SCIP 8.0 invoked through the GAMS platform. Since GUROBI can resolve the 0–1 CESP by linear approximation, we solve RTDP with GUROBI 11.0.0 directly in the experiment.

In Section 6.3, we randomly generate 150 instances for each case with $(n, L) = (60, 4)$.

6.2 Efficiency of the Algorithm

In this section, we compare the RTDP solving performance of our GB&B algorithm with the commercial solvers BARON, SCIP, and GUROBI listed in Section 4. We investigate the computational efficiency of the solution methods as n and L change. Table 2 presents the computation times of all the solution methods and the average number of branches of the GB&B algorithm.[9]

We can observe from Table 2 that all methods take more time to complete as n or L increases. It is evident that the GB&B algorithm is the most efficient one, resolving more cases in the least time. GUROBI is the least efficient, with the longest runtime and the most unsolved instances. BARON and

Table 2. Computational efficiency of different solution methods.

(n, L)		(30, 3)	(60, 3)	(30, 4)	(60, 4)	(30, 5)	(60, 5)
BARON	Max.	41.97	–	–	–	–	–
	Min.	1.23	1.96	1.38	2.81	32.48	61.22
	Avg.	5.43	161.29	85.23	219.66	312.87	517.45
	Unsolved	0	5%	10%	20%	25%	40%
SCIP	Max.	34.21	–	–	–	–	–
	Min.	0.78	1.64	0.91	1.34	20.98	16.31
	Avg.	4.32	119.12	145.38	187.62	423.19	862.28
	Unsolved	0	10%	10%	15%	35%	65%
GUROBI	Max.	–	–	–	–	–	–
	Min.	0.65	1.88	2.66	11.45	1.75	241.29
	Avg.	7.37	378.25	438.87	552.71	1,085.58	1,927.33
	Unsolved	5%	10%	5%	35%	40%	80%
Geometric branch and bound (GB&B) algorithm	Max.	3.94	4.98	5.12	7.49	9.12	17.41
	Min.	0.12	0.48	0.55	1.64	1.84	2.26
	Avg.	0.74	1.58	2.32	3.81	4.25	9.17
	Branches	2.65	4.40	9.15	13.10	30.85	33.60
	Unsolved	0	0	0	0	0	0

Table 3. Computational efficiency of the geometric branch and bound (GB&B) algorithm as (n, L) changes.

(n, L)	(120, 6)	(360, 6)	(600, 6)	(120, 8)	(360, 8)	(600, 8)	(120, 10)	(360, 10)
Max.	20.88	408.56	515.07	611.21	1,902.95	3,129.46	2,498.56	–
Min.	11.39	78.19	87.41	47.99	758.92	1,251.17	659.19	1,122.44
Avg.	19.78	305.14	526.11	171.99	1,284.18	2,544.43	1,899.73	3,069.68
Branches	58.20	91.60	118.15	422.45	518.20	732.00	3,892.90	4,283.20
Unsolved	0	0	0	0	0	0	0	10%

SCIP are slightly more efficient than GUROBI but still have many unsolved instances.

Since the GB&B algorithm is the most efficient one according to the aforementioned results, we conduct further studies on its performance on large-scale RTDP instances. We present the results in Table 3.

From Table 3, we can see that the GB&B algorithm can solve all instances of $(n, L) = (600, 8)$ and $(n, L) = (120, 10)$ within one hour. Moreover, we find that the efficiency of the GB&B algorithm is affected more by the number of sessions L than by the number of activities n. This is a typical property of GB&B algorithms, which can be found in other studies like Scholz (2012), Schöbel and Scholz (2014), and Li and Qi (2022). The reason is that the number of dimensions in the GB&B algorithm grows quickly as the number of sessions L increases, which makes the problem with a larger L difficult to solve.

6.3 *Optimal Structures of the Proportional Case*

By the literature (e.g., Das Gupta et al., 2016; Dixon and Verma, 2013; Li et al., 2023; Roels, 2020), the structure of the optimal schedule is a crucial factor for the service designer, which affects the customer experience. Therefore, we investigate the optimal structure based on the optimal solution computed by the GB&B algorithm.

According to Li et al. (2023), the structural property of the proportional case in which the attributes are linearly correlated is a key design factor. Hence, we assume that the reward, difficulty, and value of an activity are proportional, where $r_i = h_1 d_i$ and $v_i = h_2 d_i$. Here, $h_1 > 0$ and $h_2 > 0$ are the reward-difficulty and value-difficulty ratios, respectively. In this proportional case, we can describe the optimal structure without specifying the attribute, because the structures are the same for any attributes in such a circumstance. For example, if the difficulty is increasing in the optimal sequence, then the reward and value must be increasing as well, since they are proportional.

The same as Roels (2020), we focus on five structures: increasing, decreasing, U-shape, inverted U-shape (IU-shape), and pulsed. *Increasing* and *decreasing* refer to a monotonically increasing or decreasing sequence, respectively. *U-shape* refers to a decreasing subsequence followed by an increasing subsequence, while *IU-shape* refers to an increasing subsequence followed by a decreasing subsequence. *Pulsed* refers to all other sequences that do not belong to these four structures. Because the schedule of activities in each session affects the customer's participation decision in the next session, the service designer places great emphasis on how to schedule activities in each session. Therefore, we investigate the structure of each session rather than the structure of the whole

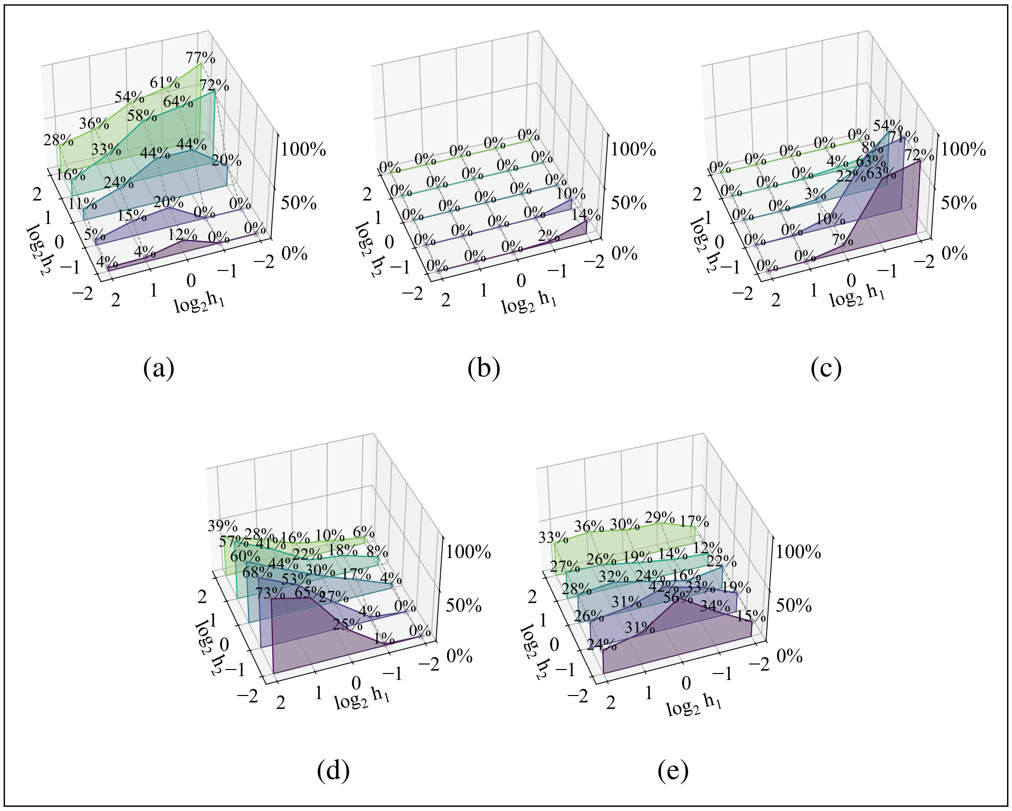

Figure 3. Percentage of per-session optimal structures with variable h_1 and variable h_2. (a) Increasing, (b) decreasing, (c) IU-shape, (d) U-shape and (e) pulsed.

training program as discussed in the interactive service design literature (e.g. Li et al., 2023; Roels, 2020).

We present the percentage of the per-session optimal structure with variable h_1 and h_2 in Figure 3.

We can tell from Figure 3 that when h_2 is large and h_1 is small, most optimal schedules are increasing. In the case where the value of each activity is high, the instantaneous functional utility dominates the instantaneous experiential utility. Because customers value the functional utility of the recent activities more than that of the previous ones, the increasing subsequence in each session is optimal under such a condition. This structure can be found in many training programs in practice when the value is highly prioritized. For example, in rehabilitation training programs, the training program usually exhibits an increasing training intensity (Eitzen et al., 2010).

When both h_1 and h_2 are small, there are more decreasing and inverted U-shape (IU)-shaped optimal subsequences. In this circumstance, the difficulty of each activity is high, and the disutility introduced by the difficulty attribute dominates. These two structures are also proposed by Li et al. (2023). This result reveals that when difficulty is high, the program will become a tough training process. Therefore, it is better to assign low-difficulty activities at both the beginning and the end of the session to provide a warm-up and cool-down for the training participants. Thus, the IU-shaped subsequence is optimal in such a circumstance. In addition, as Li et al. (2023) suggests, IU-shaped sequences may degenerate to decreasing sequences in some cases, leading to more decreasing sequences.

When h_1 is large, each activity has a high reward, and the utility introduced by the reward attribute dominates in this case. That results in a large proportion of U-shaped optimal schedules, which is similar to the case in Das Gupta et al. (2016). In this circumstance, the training program can be treated as an interactive entertainment service. For instance, during a hiking trip, the pleasure obtained from the scenery is far greater than the physical fatigue of climbing the hills. According to Das Gupta et al. (2016) and Li et al. (2022), the optimal structure of such an entertainment service is U-shaped.

We report the optimal structure and its scenario in Table 4.

7 Extensions

In this section, we discuss two extended problems of RTDP. In Section 7.1, we consider the case where there are breaks between sessions in RTDP. In Section 7.2, we investigate a scenario where the service designer cares about both customer retention and training performance.

Table 4. A summary of the optimal structure and its scenario.

h_1	h_2	Scenario	Optimal structure
Low	High	When the value is high	Increasing
Low	Low	When the difficulty is high	IU-shape or decreasing
High	Low or high	When the reward is high	U-shape

7.1 RTDP with Session Break

In the practice of training program design, breaks often occur between sessions to allow participants to relax and recharge. For instance, in a high-intensity interval training workout, there is typically a 2-3 minute rest period after every 15-minute training session.[10] In practice, the service designer also needs to optimize customer retention in this scenario. Therefore, we investigate RTDP with breaks between sessions in this section.

Similar to the identical duration τ in Section 3, we assume that each break between sessions has the same duration, denoted as $K = \kappa\tau$, where the coefficient $\kappa > 0$. We denote the remembered holistic utility in this scenario as $H_l'(\boldsymbol{\pi})$.

Similar to (10) to (12), we introduce functions $F_1'(\boldsymbol{\pi}|k)$, $F_2'(\boldsymbol{\pi}|k)$ and $F_3'(\boldsymbol{\pi}|k)$, given by

$$\begin{aligned} F_1'(\boldsymbol{\pi}|k) &= \sum_{i=1}^{N}\sum_{j=1}^{k}\pi_{i,j}r_i\Bigg(\frac{e^{-\alpha(k-j+1+\lfloor (j-1)/b\rfloor\kappa)\tau}-e^{-\omega(k-j+1+\lfloor (j-1)/b\rfloor\kappa)\tau}}{\omega-\alpha} \\ &\quad -\frac{e^{-\alpha(k-j+\lfloor (j-1)/b\rfloor\kappa)\tau}-e^{-\omega(k-j+\lfloor (j-1)/b\rfloor\kappa)\tau}}{\omega-\alpha}\Bigg), \end{aligned} \tag{40}$$

$$\begin{aligned} F_2'(\boldsymbol{\pi}|k) &= \sum_{i=1}^{N}\sum_{j=1}^{k}\pi_{i,j}d_i\Bigg(\frac{e^{-\beta(k-j+1+\lfloor (j-1)/b\rfloor\kappa)\tau}-e^{-\omega(k-j+1+\lfloor (j-1)/b\rfloor\kappa)\tau}}{\omega-\beta} \\ &\quad -\frac{e^{-\beta(k-j+\lfloor (j-1)/b\rfloor\kappa)\tau}-e^{-\omega(k-j+\lfloor (j-1)/b\rfloor\kappa)\tau}}{\omega-\beta}\Bigg), \end{aligned} \tag{41}$$

$$\begin{aligned} F_3'(\boldsymbol{\pi}|k) &= \sum_{i=1}^{N}\sum_{j=1}^{k}\pi_{i,j}v_i\Bigg(\frac{1-e^{-\omega(k-j+1+\lfloor (j-1)/b\rfloor\kappa)\tau}}{\omega} \\ &\quad -\frac{1-e^{-\omega(k-j+\lfloor (j-1)/b\rfloor\kappa)\tau}}{\omega}\Bigg). \end{aligned} \tag{42}$$

With (40) to (42), the remembered holistic utility $H_l'(\boldsymbol{\pi})$ can be expressed as

$$H_l'(\boldsymbol{\pi}) = \delta_e F_1'(\boldsymbol{\pi}|B_l) - \delta_e F_2'(\boldsymbol{\pi}|B_l) + \delta_f F_3'(\boldsymbol{\pi}|B_l). \tag{43}$$

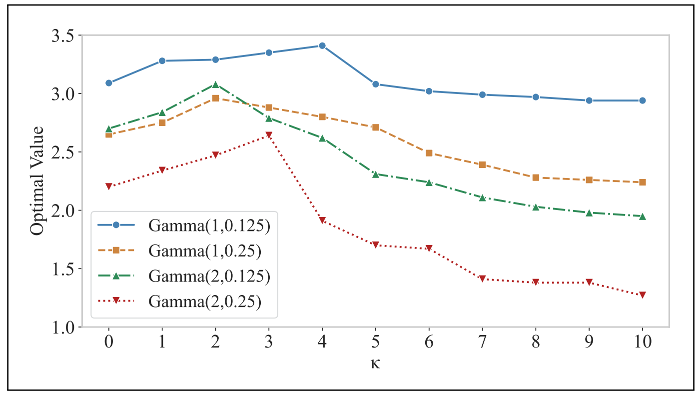

Figure 4. Optimal value of retention-oriented training program design problem with session break (RTDPB) with variable κ.

The RTDP with session break (RTDPB) is given by

$$\max_{\boldsymbol{\pi}} \Pi'(\boldsymbol{\pi}) = \sum_{l=1}^{L} w_l \prod_{i=1}^{l}(1-e^{-\lambda H_i'(\boldsymbol{\pi})}), \tag{44}$$

$$\text{s.t. Constraints (15)} - (17), (43). \tag{45}$$

By our analysis, RTDPB is an NP-hard problem, and we can solve it by a revised GB&B algorithm. The computational complexity and the solution method of RTDPB are discussed in E-companion EC.2.1.1.

To investigate the optimal value of REDPB with variable κ, we conduct numerical studies with $w_i = 1$ for all $i \in \{1, \ldots, L\}$, $k_G \in \{1, 2\}$ and $\theta_G \in \{0, 125, 0.25\}$. Similar to Section 6.3, we randomly generate 150 instances with $(n, L) = (60, 4)$ for each case. The other experiment setups are the same as in Section 6.1. We present the optimal value with variable κ in Figure 4. The detailed data in Figure 4 are shown in E-companion EC.2.1.2.

Figure 4 indicates that as κ increases, the optimal value first increases and then decreases. This result aligns well with practical experience. A medium break duration can improve customer retention, while excessively long or short breaks can lead to a decline in retention. The reason is that if the break duration is short, consecutive sessions can maintain the customer's reference level at a high degree. This can reduce customer instantaneous and remembered holistic utility, leading to a decline in customer retention. If the break duration is long, customers may forget previous experiences, leading to a reduction in the customer's remembered holistic utility and retention. Therefore, it is essential for the service designer to choose an appropriate length for break durations to optimize customer retention.

7.2 Joint Retention-Performance Optimization

In RTDP, as discussed in Section 3.3, the service provider only cares about the customer retention of the training program, ignoring the performance of customers. However, in practice, the service provider may hope to optimize both the customer retention and performance of a training program. Therefore, in this section, we study a joint retention-performance optimization problem.

Following Banister et al. (1975) and Roels (2020), the performance of the customer depends on fitness and fatigue levels. Fitness (level) refers to the customer's physical condition, including endurance, strength, and skill. Since fitness is related to the functional value of activities, fitness is derived from the value attribute. Fatigue is a common result of the training activities. As discussed in Section 3.1.1, fatigue is derived from the difficulty attribute. The training process can enhance the fitness level and reduce the fatigue level.

According to Roels (2020), the fitness level at the end of the training program depends on the adaptation and memory decay process. The adaptation process reflects the phenomenon where fitness will no longer improve if the customer participates in activities with the same functional value consecutively. The adaptation process of the fitness is similar to what was discussed in Section 3.1.1.

Similar to (3) and (4), the instantaneous fitness level $u_v(t)$ at $t \in [t_{i-1}, t_i]$ is given by

$$u_v(t) = v_{(i)} - f_v(t),$$

where $f_v(t)$ refers to the fitness reference level at time t.

Let $\gamma > 0$ denote the adaptation rate of the fitness. Similar to (5) and (6), the change rate of reference fitness level $f_v(t)$ is given by

$$\frac{df_v(t)}{dt} = \gamma(v_{(i)} - f_v(t)). \tag{46}$$

With (46), the instantaneous fitness level at $t \in [t_{i-1}, t_i]$ can be expressed as

$$u_v(t) = ((v_{(1)} - f_v(0)) + \sum_{j=2}^{i}(v_{(j)} - v_{(j-1)})e^{\gamma t_{j-1}})e^{-\gamma t}. \tag{47}$$

The memory decay process depicts that fitness will gradually decay due to forgetting (Roels, 2020), which is similar to what was discussed in Section 3.1.2. We denote the remembered fitness level at the end of the training program as $F_4(\boldsymbol{\pi}|n)$, which is a stock of the instantaneous fitness level, expressed as

$$F_4(\boldsymbol{\pi}|n) = \sum_{i=1}^{n}\int_{t_{i-1}}^{t_i} u_v(t)e^{-\omega(t_n - t)}dt. \tag{48}$$

Combining (47) and (48), we have

$$F_4(\boldsymbol{\pi}|n) = \sum_{i=1}^{N}\sum_{j=1}^{n}\pi_{i,j}v_i\left(\frac{e^{-\gamma(k-j+1)\tau} - e^{-\omega(k-j+1)\tau}}{\omega - \gamma} - \frac{e^{-\gamma(k-j)\tau} - e^{-\omega(k-j)\tau}}{\omega - \gamma}\right) - f_v(0)\frac{e^{-\gamma k\tau} - e^{-\omega k\tau}}{\omega - \gamma}. \tag{49}$$

We adopt the multiplicative fitness-fatigue model proposed in Roels (2020), which is suitable for the skill retention period. During this period, the customer has already had some previous training experience, which matches typical retention-oriented training scenarios. Recall what we defined in (11) of Section 3.1.2, the fatigue level at the end of the training program is $F_2(\boldsymbol{\pi}|B_L)$. The performance of a customer if he completes the training program with L sessions (a.k.a., the retained performance) is given by

$$P_L(\boldsymbol{\pi}) = P_0 + \delta_p\frac{F_4(\boldsymbol{\pi}|B_L)}{F_2(\boldsymbol{\pi}|B_L)}, \tag{50}$$

where P_0 denotes the initial performance at the beginning of the training program, $\delta_p > 0$ is a performance coefficient, and B_L is the number of activities in L sessions as defined in Section 3. In practice, because the overall fitness and fatigue of a training program are positive, we restrict $F_2(\boldsymbol{\pi}|B_L) > 0$ and $F_4(\boldsymbol{\pi}|B_L) > 0$.

When the service provider cares about both retention and retained performance, the JTOP is given by

$$\max_{\boldsymbol{\pi}} \Pi_J(\boldsymbol{\pi}) = \sigma\sum_{l=1}^{L} w_l\prod_{i=1}^{l}(1 - e^{-\lambda H_i(\boldsymbol{\pi})}) + (1-\sigma)P_L(\boldsymbol{\pi}), \tag{51}$$

$$\text{s.t. Constraints (49) and (50),} \tag{52}$$

$$F_2(\boldsymbol{\pi}|B_L) > 0, \tag{53}$$

$$F_4(\boldsymbol{\pi}|B_L) > 0, \tag{54}$$

$$\boldsymbol{\pi} \in \mathcal{T}, \tag{55}$$

where $\sigma \in [0, 1]$ reflects the preference of the service designer.

The additive formulation (51) reflects JTOP is a multi-objective optimization problem that captures the designer's dual objective to optimize both the retention and retained performance of a training program. When $\sigma = 1$, the service designer is only concerned about customer retention as in Section 3, and JTOP becomes RTDP in this circumstance. When $\sigma = 0$, the service designer cares only about the performance of a retained customer who completes the whole program, capturing the case of the performance-oriented training program design (e.g., Roels, 2020). When $\sigma \in (0, 1)$, the service designer cares about both customer retention and retained performance.

By our analysis, JTOP is an NP-hard problem, and it can be solved by a modified GB&B algorithm. We discuss the computational complexity and the solution method of JTOP

Table 5. Percentage of per-session optimal structures with variable σ.

Structure	Increasing	Decreasing	U-shape	IU-shape	Pulsed
$\sigma = 0$	0.67%	0%	0%	0%	99.33%
$\sigma = 0.5$	9.66%	1.33%	5.33%	4.00%	79.67%
$\sigma = 1$	46.67%	7.33%	16.67%	16.00%	13.33%

in E-companion EC.2.2.1. We also conduct a sensitivity analysis on the efficiency of the modified GB&B algorithm in E-companion EC.2.2.2.

We then investigate the structural properties of the optimal schedule based on the optimal solution computed by the modified GB&B algorithm and discuss the insights for training program design. If not specifically mentioned, the setups of the numerical experiments in this section are the same as those discussed in Section 6.1. We conduct a numerical experiment with $\sigma \in \{0, 0.5, 1\}$ and $f_v(0) = 0$. Similar to Section 6.3, we randomly generate 150 instances for the case with $(n, L) = (60, 4)$, where $r_i = h_1 d_i$ and $v_i = h_2 d_i$. The ratio parameters h_1 and h_2 are generated randomly from the uniform distribution Uniform(0,2).

We present the percentage of the per-session optimal structures in Table 5.

Table 5 illustrates that as σ increases, there are less pulsed subsequences. In the case where the service designer only cares about customer performance ($\sigma = 0$), nearly all (99.33%) subsequences are pulsed. This finding is consistent with Daniels's marathon training plan (Daniels, 2022). This plan recommends a training program with repeatedly switched training intensities, presenting a pulsed intensity sequence. The reason is that the pulsed sequence with frequently switched low- and high-intensity exercises helps improve the training performance efficiently (Eliaz and Spiegler, 2021). When the service designer cares only about customer retention ($\sigma = 1$), most of the subsequences are nonpulsed. The reason is that these nonpulsed structures (i.e., increasing, decreasing, U-shaped, and IU-shaped sequences) help enhance customer experience, resulting in increased customer retention. These structures are also suggested by the experiential service design literature (e.g. Das Gupta et al., 2016; Li et al., 2023), which optimizes customer satisfaction. When the service designer is concerned about both customer retention and performance ($\sigma = 0.5$), the percentage of the pulsed optimal subsequence is at an intermediate level.

To illustrate the optimal structure of the training program in different situations, we analyze the optimal sequences of an example case with the same set of activities. In this example, 60 activities are split into 4 sessions, containing 15 activities. The parameters of the example and optimal schedules are listed in E-companion EC.2.2.3. We illustrate the optimal structures in Figure 5.

We can tell from Figure 5 that when $\sigma = 0$, the optimal sequence in each session (each session has 15 activities) is pulsed, with dramatic changes between adjacent elements (see Figure 5(a)). This training plan that alternates between high- and low-intensity exercises enhances the training performance (Eliaz and Spiegler, 2021). In contrast, when $\sigma = 1$, the optimal sequence in each session is IU-shaped, with relatively smooth changes between adjacent elements (see Figure 5(c)). This sequence can alleviate the stress and fatigue of customers, which improves customer retention. When $\sigma = 0.5$ (see Figure 5(b)), although the sequence in each session is pulsed, the changes in difficulty levels are less severe compared to the performance-oriented training shown in Figure 5(a).

8 Conclusion

The training industry is an important sector in the service economy that attracts the attention of both service design practitioners and researchers. Although it is crucial to participate in all training sessions, many customers fail to complete a program due to fatigue and stress. Due to the significance of training retention optimization, we investigate the design of the retention-oriented training program, discussing both a model of customer retention and a method to find the optimal training schedule.

We consider the behavior that a customer may abandon the training program, which is common in practice but overlooked by the training design literature. Similar to Aflaki and Popescu (2014), the customer makes the participation decision based on the remembered holistic utility, and we consider the case where that utility depends linearly on experiential and functional utilities, as discussed in Bellos and Kavadias (2021). We formulate the RTDP, which optimizes the retention of a training program by scheduling training activities.

By our analysis, RTDP is a 0–1 CESP, which we show is NP-hard. This problem is typically solved by the B&B technique, which relies on certain properties of the decision problem to compute bounds on the objective function. However, RTDP lacks these desirable properties. As an alternative, we develop a GB&B algorithm inspired by Scholz (2012), which computes the bound of the objective function by solving a series of subproblems. Our numerical studies suggest that our GB&B algorithm is more efficient than other solution methods that can be used in service design practice. Then we discuss the structural properties and insights into retention-oriented training program design in the proportional case. We find that higher reward, difficulty, and value result in more U-shaped, IU-shaped, and increasing subsequences, respectively.

We also investigate several extensions of RTDP. First, we discuss RTDP with session break. Our finding reveals that as the break duration increases, customer retention changes in a nonmonotonic fashion. Second, we investigate the JTOP, where there are more pulsed subsequences when the designer cares more about the performance, and fewer pulsed subsequences when the focus on retention increases.

We believe that future studies should explore several related topics that are beyond the scope of our study. First, similar to Bellos and Kavadias (2021) and Li et al. (2023), we assume

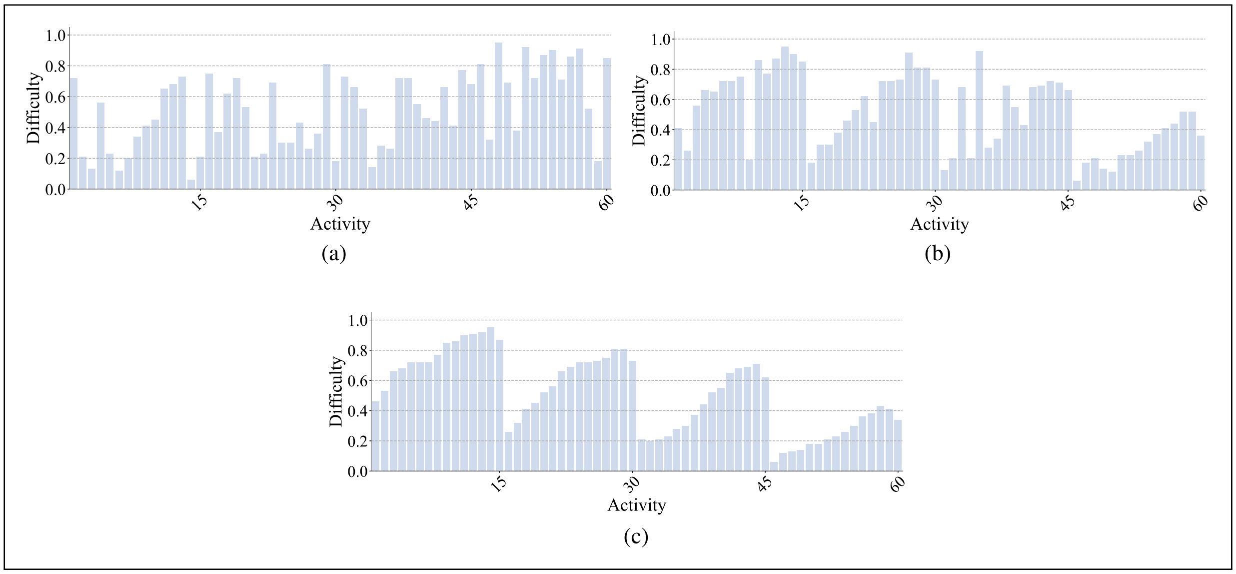

Figure 5. Optimal structures of the example case with $(n, L) = (60, 4)$ and Variable σ. (a) Optimal sequence with $\sigma = 0$ (performance-oriented training), (b) optimal sequence with $\sigma = 0.5$ (joint training) and (c) optimal sequence with $\sigma = 1$ (retention-oriented training).

the remembered holistic utility and instantaneous experiential utility are in linear form. Future research could consider these utilities in nonlinear forms. Second, in our work, we consider the adaptation process where the instantaneous disutility of a fixed-difficulty activity decays overtime, consistent with Li et al. (2023). This reflects that the customer gradually adapts to a given difficulty level. However, in practice, if the duration of an activity is excessively long, the customer may become increasingly fatigued at the end of the activity due to physical limitations. This may lead to a more complicated adaptation process. It would be interesting to investigate the service design problem with such an adaptation process. Third, similar to Aflaki and Popescu (2014), we consider the scenario where customers make their participation decision based on previous experience. Although this fits the training practice well, certain customers may make decisions based on their expectations regarding future service. It would be interesting to consider such a scenario where customers are forward-looking. Fourth, consistent with Das Gupta et al. (2016), we consider the adaptation and memory decay effects in customer holistic remembered utility without capturing the satiation effect. Future research could explore how to design a training program considering this effect (e.g., Baucells and Zhao, 2020). Fifth, like Roels (2020), we investigate the training design problem within a finite horizon. Given that certain training programs can be a life-long practice (e.g., the rehabilitation training programs discussed in Langhorne et al. (2011)), it would be interesting to investigate the training design problem with an infinite horizon. Finally, in JTOP, we investigate a bi-objective optimization problem that maximizes both customer retention and performance. In some circumstances, the designer may care about optimizing the cumulated performance conditional on the retention rate in each session (i.e., $\sum_{l=1}^{L} \hat{R}_l(\boldsymbol{\pi})P_l(\boldsymbol{\pi})$). It is interesting to investigate the solution approach to the training design problem under such a situation.

Acknowledgments

The authors thank the department and senior editors, and two anonymous reviewers for the constructive review process that helped improve the paper. Authors are listed in alphabetical order and share the first authorship.

Declaration of Conflicting Interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The authors received the following financial support for the research, authorship, and/or publication of this article: This research is supported by the National Natural Science Foundation of China Projects [72201210,72471216], and the USTC Research Funds of the Double First-Class Initiative Project [YD2040002026].

ORCID iDs

Qiuwei Guo https://orcid.org/0009-0007-1401-4058
Yifu Li https://orcid.org/0000-0002-0040-1205
Lindong Liu https://orcid.org/0000-0003-1015-3097
Lifei Sheng https://orcid.org/0000-0003-3592-0619

Supplemental Material

Supplemental material for this article is available online (doi: 10.1177/10591478251369160).

Notes

1. See https://runrepeat.com/fitness-industry (accessed: Jul 30, 2024).

2. See https://www.who.int/news-room/fact-sheets/detail/rehabilitation (accessed: Jul 30, 2024).
3. See https://inbodyusa.com/blogs/inbodyblog/what-happens-when-you-stop-working-out/ for the difficulties in restarting training (accessed: Jul 30, 2024).
4. See https://www.precedenceresearch.com/home-rehabilitation-products-and-services-market (accessed: May 17, 2025).
5. See https://apps.apple.com/us/app/7-minute-workout/id650762525 (accessed: Jul 30, 2024).
6. See https://www.ohio.edu/recreation/fitness/f45 for the F45 Training (accessed: Jul 30, 2024).
7. In our study, we consider the retention-oriented programs with a finite planning horizon (i.e., L is a finite number in our study). In practice, there can be certain programs with an infinite horizon, in which the number of training sessions is infinite. An example is stroke rehabilitation training, which involves the slow and prolonged process of neural repair and requires sustained training in a life-long timeframe (Langhorne et al., 2011). There can be a series of subscription cycles in these programs, each with multiple training sessions. For example, there could be multiple courses of treatment in the rehabilitation training program, with each course lasting for weeks or months, which can be treated as a subscription cycle with multiple training sessions. The doctor can adjust the program design at the beginning of each course of treatment. Given this situation, our study can be treated as a single subscription cycle design problem with L sessions.
8. See https://docs.gurobi.com/projects/optimizer/en/current/features/nonlinear.html for details of the piecewise-linear approximation method in GUROBI (accessed: Apr 13, 2025).
9. Throughout our numerical studies, we use "max", "min", and "avg" to represent the maximum, minimum, and average computation time for all instances, respectively. We report them in seconds. "Unsolved" indicates the percentage of instances that could not be solved within 1 hour. "Branches" refers to the average number of times the box is cut or divided to obtain the optimal solution.
10. See https://runningonrealfood.com/45-minute-no-equipment-hiit-workout/ (accessed: Apr 1, 2025).

References

Aflaki S and Popescu I (2014) Managing retention in service relationships. *Management Science* 60(2): 415–433.

Alptekinoğlu A and Semple JH (2016) The exponomial choice model: A new alternative for assortment and price optimization. *Operations Research* 64(1): 79–93.

Banister EW, Calvert TW, Savage MV, et al. (1975) A systems model of training for athletic performance. *Australian Journal of Sports Medicine* 73(3): 57–61.

Baucells M and Sarin RK (2007) Satiation in discounted utility. *Operations Research* 55(1): 170–181.

Baucells M and Sarin RK (2010) Predicting utility under satiation and habit formation. *Management Science* 56(2): 286–301.

Baucells M and Zhao L (2019) It is time to get some rest. *Management Science* 65(4): 1717–1734.

Baucells M and Zhao L (2020) Everything in moderation: Foundations and applications of the satiation model. *Management Science* 66(12): 5701–5719.

Bellos I and Kavadias S (2019) When should customers control service delivery? Implications for service design. *Production and Operations Management* 28(4): 890–907.

Bellos I and Kavadias S (2021) Service design for a holistic customer experience: A process framework. *Management Science* 67(3): 1718–1736.

Ben Rhouma T and Zaccour G (2018) Optimal marketing strategies for the acquisition and retention of service subscribers. *Management Science* 64(6): 2609–2627.

Bernstein F, Chakraborty S and Swinney R (2022) Intertemporal content variation with customer learning. *Manufacturing & Service Operations Management* 24(3): 1664–1680.

Bestuzheva K, Chmiela A, Müller B, et al. (2025) Global optimization of mixed-integer nonlinear programs with scip 8. *Journal of Global Optimization* 91: 287–310.

Chen M, Elmachtoub A and Lei X (2021) Matchmaking strategies for maximizing player engagement in video games. *Working paper* Available at SSRN 3928966.

Chen H, Hu M, Liu J, et al. (2024) Ups and downs in experience design. *Production and Operations Management* 33(9): 1895–1911.

Daniels J (2022) *Daniels' Running Formula*. 4rd ed edition. Champaign, IL: Human Kinetics.

Das Gupta A, Karmarkar US and Roels G (2016) The design of experiential services with acclimation and memory decay: Optimal sequence and duration. *Management Science* 62(5): 1278–1296.

De Bruecker P, Beliën J, Van den Bergh J, et al. (2018) A three-stage mixed integer programming approach for optimizing the skill mix and training schedules for aircraft maintenance. *European Journal of Operational Research* 267(2): 439–452.

Deshmane A, Martínez-de Albéniz V and Roels G (2023) Intertemporal spillovers in consumer experiences: Empirical evidence and service design implications. *Working paper* Available at SSRN 4507191.

Dixon MJ and Thompson GM (2016) Bundling and scheduling service packages with customer behavior: Model and heuristic. *Production and Operations Management* 25(1): 36–55.

Dixon M and Verma R (2013) Sequence effects in service bundles: Implications for service design and scheduling. *Journal of Operations Management* 31(3): 138–152.

Eason J (2013) New study finds 73 percent of people who set fitness goals as new year's resolutions give them up. *Bodybuilding* Available at: https://www.bodybuilding.com/fun/2013-100k-transformation-contest-press-release.html (accessed date 30 July 2024).

Ebbinghaus H (2013) Memory: A contribution to experimental psychology. *Annals of Neurosciences* 20(4): 155.

Eitzen I, Moksnes H, Snyder-Mackler L, et al. (2010) A progressive 5-week exercise therapy program leads to significant improvement in knee function early after anterior cruciate ligament injury. *Journal of Orthopaedic & Sports Physical Therapy* 40(11): 705–721.

Eliaz K and Spiegler R (2021) Anabolic persuasion. Working paper Available at arXiv 2105.08786.

Elloumi S, Lambert A and Lazare A (2021) Solving unconstrained 0-1 polynomial programs through quadratic convex reformulation. *Journal of Global Optimization* 80(2): 231–248.

Forrest NC, Hill RR and Jenkins PR (2022) An air force pilot training recommendation system using advanced analytical methods. *INFORMS Journal on Applied Analytics* 52(2): 198–209.

Hall NG and Liu Z (2023) Scheduling with present bias. *Production and Operations Management* 32(6): 1743–1759.

Icmeli O and Erenguc SS (1996) A branch and bound procedure for the resource constrained project scheduling problem with discounted cash flows. *Management Science* 42(10): 1395–1408.

Kanoria Y, Lobel I and Lu J (2024) Managing customer churn via service mode control. *Mathematics of Operations Research* 49(2): 1192–1222.

Ladany SP (1975) Optimization of pentathlon training plans. *Management Science* 21(10): 1144–1155.

Langhorne P, Bernhardt J and Kwakkel G (2011) Stroke rehabilitation. *The Lancet* 377(9778): 1693–1702.

Lei X, Wan B and Wang S (2023) Content rotation in the presence of satiation effect. *Working paper* Available at SSRN 4593945.

Li Y, Dai T and Qi X (2022) A theory of interior peaks: Activity sequencing and selection for service design. *Manufacturing & Service Operations Management* 24(2): 993–1001.

Li Y and Qi X (2022) A geometric branch-and-bound algorithm for the service bundle design problem. *European Journal of Operational Research* 303(3): 1044–1056.

Li Y, Ryan CT and Sheng L (2023) Optimal sequencing in single-player games. *Management Science* 69(10): 6057–6075.

Li Y, Sheng L and Zhao X (2024) Player game design: Boost engagement in contests through intervention. Working paper available at SSRN 4948548.

Liao C and Chen Y (2021) Design of long-term conditional cash transfer program to encourage healthy habits. *Production and Operations Management* 30(11): 3987–4003.

Long X, Nasiry J and Wu Y (2020) A behavioral study on abandonment decisions in multistage projects. *Management Science* 66(5): 1999–2016.

Mittelmann H (2021) Mixed integer nonlinear programming benchmark (minlplib). *Internet Archive* Available at: https://web.archive.org/web/20211220190643/http://plato.asu.edu/ftp/minlp.html (accessed date 30 July 2024).

Mohan S (2024) 3 day muscle maintenance workout plan. *Dr Workout* Available at: https://www.drworkout.fitness/muscle-maintenance-workout-plan/ (accessed date 30 July 2024).

Peek K, Carey M, Mackenzie L, et al. (2018) Patient adherence to an exercise program for chronic low back pain measured by patient-report, physiotherapist-perception and observational data. *Physiotherapy Theory and Practice* 35(12): 1–10.

Qi X, Bard JF and Yu G (2004) Class scheduling for pilot training. *Operations Research* 52(1): 148–162.

Roels G (2014) Optimal design of coproductive services: Interaction and work allocation. *Manufacturing & Service Operations Management* 16(4): 578–594.

Roels G (2020) High-performance practice processes. *Management Science* 66(4): 1509–1526.

Schöbel A and Scholz D (2014) A solution algorithm for non-convex mixed integer optimization problems with only few continuous variables. *European Journal of Operational Research* 232(2): 266–275.

Scholz D (2012) *Deterministic Global Optimization: Geometric Branch-and-bound Methods and Their Applications*. New York, NY: Springer Science & Business Media.

Ursulenko O, Butenko S and Prokopyev OA (2013) A global optimization algorithm for solving the minimum multiple ratio spanning tree problem. *Journal of Global Optimization* 56(3): 1029–1043.

Vanhoucke M, Demeulemeester E and Herroelen W (2001) On maximizing the net present value of a project under renewable resource constraints. *Management Science* 47(8): 1113–1121.

Wang H (1995) A branch-and-bound approach for sequencing expansion projects. *Production and Operations Management* 4(1): 57–75.

Zhao W, Hall NG and Liu Z (2020) Project evaluation and selection with task failures. *Production and Operations Management* 29(2): 428–446.

Zhu Y, Liu Z and Qi X (2024) Nonsequential appointment scheduling with a random number of requests. *Production and Operations Management* 33(1): 184–204.

Zwols Y and Sierksma G (2009) Training optimization for the decathlon. *Operations Research* 57(4): 812–822.

How to cite this article

Guo Q, Li Y, Liu L and Sheng S (2025) Maximizing Customer Retention in Multisession Training Service: Model and Algorithm. *Production and Operations Management* XX(XX): 1–20.