Linear Search or Binary Search

Guo Bai *

November 16, 2022

Abstract

I investigate how temporal risk attitude and time risk attitude affect an agent's optimal search strategy within a dynamic single-agent's learning model. To discuss the agent's time risk attitude, a generalised lifetime utility is considered. I show that when the agent's prior belief is uniform and the preference is time-consistent, only Linear Search and Binary Search can be optimal search strategies. No other behaviour is optimal.

Keywords: time risk, time preferences, search

JEL Codes: D81, D83, D91

1 Introduction

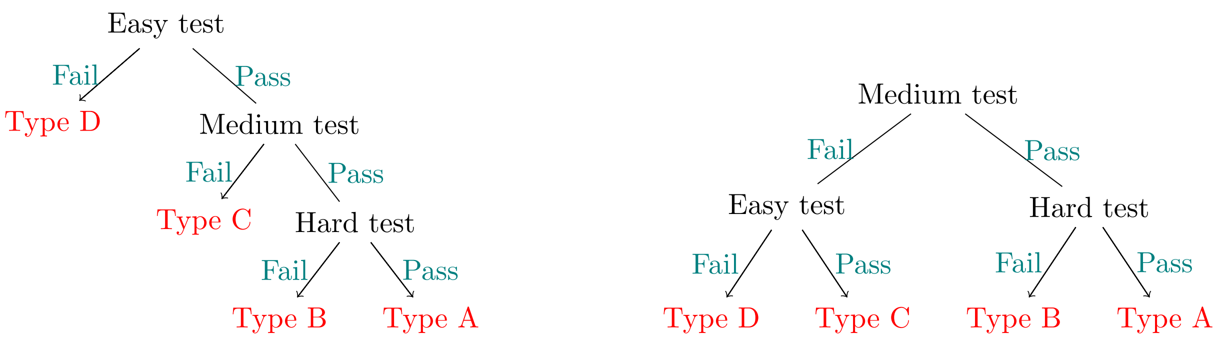
In order to learn, people search for information in different ways. Imagine yourself searching for a word in the dictionary. Do you search the page from top to bottom, or check the middle first? When searching from top to bottom, each candidate is checked sequentially. I will call this *Linear Search*. If instead, you were to check first in the middle, half the candidates would be eliminated immediately. I will call this *Binary Search*. [1]

Consider the following example of a search problem. An employer wants to learn about the new employee so that appropriate tasks can be assigned to her in the future. The employer believes that the employee can be one of four types: A, B, C and D. The type A employee is the most capable. Type B is more capable than type C, and type D is the least capable. The employer learns the employee's type by giving her some tests. There are three kinds of tests: easy ones, medium ones, and hard ones. The type A passes all tests, while the type B only fails the hard tests. The type C passes the easy tests, while the type D cannot pass any. After choosing the test, the employer observes whether the employee passed or not. The employer may ask the employee to do as many tests as required until she learns the employee's type. Throughout, I will assume that the employee does not know that the tests are designed to learn her type; the employee is not strategic and will always try her best to pass.

---

*PhD student, Department of Economics, University College London. Address: Drayton House, 30 Gordon Street, London, WC1H 0AX. Email: guo.bai.15@ucl.ac.uk. I am grateful to Martin Cripps, Deniz Kattwinkel, Konrad Mierendorff for supervision and support. I would also like to thank Antonio Cabrales, Gherardo Caracciolo, David Dillenberger, Amir Habibi and Nikita Roketskiy for helpful comments.

[1] Linear Search and Binary Search are standard terminologies in Computer Science literature.



(a) Linear Search (b) Binary Search

Figure 1: Demonstration of the example

The question of interest is what sequence of tests the employer should choose. Figure 1 demonstrates two possible choices. Each node represents the test of the employer's choice. The left panel is an example of a Linear Search, where the employer always chooses the easiest test such that only one of the remaining types will fail. The right panel is an example of a Binary Search. The employer chooses the test such that half the types will pass while the other half will not.

Linear Search and Binary Search both allow the agent to learn the employee's type eventually. They differ in two aspects. The first difference is the number of tests required to learn the employee's type. Binary Search allows the employer to learn after an almost fixed number of tests, [2] and Linear Search requires a variable number of tests. In the example, by doing Binary Search, the employer learns after two tests, while by doing Linear Search, she may learn after one, two, or three tests. In this paper, I assume that the employer can only do one test at each time. Linear Search is risky in the sense that the employer may learn the employee's type at different time, while Binary Search is safer, but the employer will never learn the employee's type immediately. Second, Linear Search and Binary Search differ in the expected number of tests required to learn the employee's type. In other words, the expected duration of learning. Binary Search allows the employee to learn faster in expectation. In the example, by doing Binary Search, the expected number of required tests is two, while by doing Linear Search, the expected number of required tests is 2.25. The agent learns faster in expectation by doing Binary Search.

Which sequence of tests is optimal depends on the employer's intertemporal preference. Specifically, it depends on the patience level and the time risk attitude of the employer. An impatient and time-risk loving agent will find Linear Search optimal, while a patient and time-risk averse agent will find Binary Search optimal. There is a trade-off between being patient and time-risk loving.

I discuss the tradeoff with a single-agent model where an agent searches for information to learn the true value of a parameter. In the example, the type of the employee is abstracted as an unknown parameter. The agent believes that the unknown parameter is drawn from a finite set and is fixed over time. The elements in the parameter set are sorted from the lowest to the highest. At each time the agent chooses an element to test. The test has two outcomes: pass or fail. The

[2]There is a small variation of plus and minus one depending on whether the number of types is a power of two. If the number of types is a power of two, the employer learns the type after a fixed number of tests. If it is not, the employer learns the type after m or $m + 1$ tests with positive probability.

element chosen fails the test if it is greater than the true value, and it passes otherwise. The agent chooses a sequence of tests to learn the true value of the parameter. Each test can be interpreted as a signal structure with two signals. By choosing different tests, the agent effectively chooses both the informativeness of the signals and the ex-ante probability of receiving each signal. Consider the example again. On the left panel, whenever the 'fail' signal arrives, it tells the employer everything. The 'fail' signal is very informative and it allows the employer to perfectly learn the type. The downside of this informative signal is that it does not arrive frequently. The other signal, the 'pass' signal arrives more frequently. However, when the 'pass' signal arrives, it only tells the employer something, but not everything, about the type of the employee. It helps the employer to eliminate some types, but does not allow the employer to perfectly learn the type. The properties of the signals change when the employer chooses other sequence of tests. In the example, on the right panel, both the 'fail' signal and 'pass' signal arrive with the same probability, and contains the same amount of information.

The main purpose of this paper is to discuss how different preferences affect the agent's optimal choice of the sequence of the tests. In particular, the different *time* preferences. Two different types of risk attitude will be discussed in this paper: the standard temporal risk and the time risk. The time risk is the agent's risk attitude towards time lotteries. Imagine you can choose between two lotteries. One lottery gives you a reward of one today or the day after tomorrow with equal probabilities. The other lottery gives you a reward of one tomorrow with probability one. These two lotteries offer the same expected rewards, while the former one is riskier. If one evaluates the payoff by expected utility with exponential discounting, she will always prefer the risky one. That is, the expected utility with exponential discounting describes risk-loving time risk attitude. The expected utility with exponential discounting will be considered as a benchmark case. A more general lifetime utility will be considered to discuss the effect of different time risk attitudes.

In the main text, I characterise the optimal sequence of tests given different preferences with an additional assumption that the agent has a uniform prior belief. I will show that when the agent has a uniform prior belief and a time-consistent preference, among all sequences of tests, only two of them really matter: always uses Linear Search and always uses Binary Search. That is, all other sequences of tests, for example, the agent switches between Linear Search and Binary Search, are always sub-optimal. The optimality of the sequence of Linear Search and Binary Search depends on the agent's discount parameter in the expected utility with exponential discounting case, and depends on the agent's 'virtual discount parameter', which is determined by the agent's time risk and temporal risk attitude in the cases with a more general lifetime utility.

1.1 Related Literature

This paper is closely related to Zhong (2019). Zhong (2019) discusses a general dynamic information acquisition framework where the agent can choose any information structure subject to a flow cost constraint at each time. Zhong (2019) shows that when the agent is an expected utility maximiser who discounts the future payoffs exponentially, the Poisson signal is the optimal signal structure. This is because the Poisson signal generates the learning time with the highest variance compared to other signal structures. Since the agent discounts the future payoffs exponentially, she has the risk-loving time preference, and hence prefers the signal structure that induces the high variance of learning time. The other paper Zhong (2017) explicitly discusses the relationship between the optimal information acquisition and the agent's time preference. Zhong (2017) shows that subject to a flow cost constraint, any information structure induces the same expected learning time, and Poisson signal induces the largest variance. The agent with risk-loving time preference hence prefers Poisson signal. I focus on comparing how agent obtains information when she has different preferences given a specific class of information structures, while Zhong's papers focus on discussing the agent's optimal information choices in a general framework.

The signal structure in this paper tells the agent the learning direction. It can be imagined as the indicator at the intersection of two roads that tells the agent which direction to go in order to find the target. This shares some similar features as in Callander (2011). In Callander (2011), the agent wants to learn about a mapping from the choices to the outcomes. The mapping is modelled as a realised Brownian motion, where the agent knows the parameters that characterise the Brownian motion, but does not know its realisation. In order to learn about the realised Brownian motion, the agent can observe the alternative and outcome pairs that has been chosen by his predecessors and then choose an alternative to learn its outcome. Because of the property of the Brownian motion, when the agent observes the alternative and outcome pairs of his predecessors, if the previous outcomes are not of satisfaction, the agent learns the direction to search for the next alternative. This has the similar features as the signal structure in my paper, but it is different. In Callander (2011), when the agent chooses an alternative, the agent learns the outcome associated with that alternative. However, in my paper, the signal itself does not allow the agent to learn the unknown parameter directly. The signal only serves as an indicator that tells the agent the learning direction. Learning happens when there is only one candidate left towards that direction. Learning can be considered as indirect in this sense.

This paper is related to the literature that discusses the preferences on time lotteries. This is because the final payoff associated with learning can be considered as a time lottery. Given a search method, the agent faces a time lottery that gives him a payoff of one at a random time. Dillenberger et al. (2018) and Dejarnette et al. (2018) discuss the preferences on time lotteries. One of the ideas in those papers is that the commonly used expected utility with exponential discounting describes a risk-loving time preference. In order to characterise other time risk attitude, a generalised expected utility should be considered. My paper can be considered as an application of the generalised expected utility to the learning and searching problem.

This paper is also related to the literature that discusses the timing of resolution of uncertainty. If the agent always uses Binary Search, then the uncertainty about the timing of learning is resolved, and hence it can be regarded as an early resolution of uncertainty. This can be shown from the example in the introduction. If the employer always uses Binary Search, she knows that she will learn the type of the employee at the second time period. However, if the agent always uses Linear Search, then the timing of learning the unknown remains uncertain, and it thus can be regarded as a late resolution of uncertainty. There is a group of literature discussing the preferences on early and late resolution of uncertainty, including Epstein & Zin (1989), Kreps & Porteus (1978), Dillenberger (2010) and Palacios-Huerta (1999). This paper can be considered as an application of the preferences on resolution of uncertainty to the learning and searching problem.

The last group of the literature is the computer science literature on Linear and Binary Search. In computer science, Linear Search and Binary Search are algorithms to find the position of a target value. The details of these algorithms can be found in **?**. The recent computer science papers including **?**, **?**, and **?** compare the Linear Search and Binary Search algorithms in different situations. While the computer science literature focuses on comparing the speed and the complexity of the search algorithms, this paper focuses on discussing how people's time preferences affect their optimal choices of the search algorithms. Without discussing which search algorithm allows the agent to learn faster, this paper discusses how agent's preferences, in particular, agent's patient level and the risk attitude, affect their choice of the search algorithms.

2 The model

This section first introduces the model setup and then discusses the assumptions of the model.

2.1 Setup

There is a single decision-maker in this model. I call her the *agent*. The agent wants to learn the true value of a parameter $\theta \in \Theta$, where Θ is finite with the cardinality $\bar{N}$. I assume that the set Θ is sorted such that $\theta_1 < \theta_2 < \cdots < \theta_{\bar{N}}$. The parameter θ is drawn from a distribution F with the probability mass function f at the beginning and it is fixed over time. Time $t = 0, 1, \ldots, T$ is discrete and finite. The final period T is greater than $\bar{N} - 2$. The agent's action at each time t is to choose an element $r_t \in \Theta$ to test.

The test at time t has two outcomes: pass or fail. The element chosen fails the test if it is greater than the true value, and it passes otherwise. By choosing the test at time t, the agent effectively chooses the information structure as described below. The information structure at each time t consists of a binary signal $s_t \in \{0, 1\}$ and a probability distribution over the signals. The 0 signal is the fail signal, while the 1 signal is the pass signal. Conditional on the true value of the parameter and the test chosen, the probabilities of receiving the signals are

$$\Pr(s_t = 0 \mid \theta < r_t) = \Pr(s_t = 1 \mid \theta \geq r_t) = 1.$$

A pair (r_t, s_t) is the agent's action and the signal received at time t. The agent remembers her past actions and the past signals received. Let $r^t = \{r_0, r_1, \ldots, r_{t-1}\}$ be the sequence of the past actions, and let $s^t = \{s_0, s_1, \ldots, s_{t-1}\}$ be the sequence of the past signals. At the beginning of time t, $I_t = \{r^t, s^t\}$ denotes the history up to that point. The set of histories is denoted by $\mathcal{I} = \bigcup_{t=1,2,\ldots,T} I_t \cup \emptyset$. The strategy of the agent is given by a mapping $\mathcal{R} : \mathcal{I} \to \Theta$ from the histories to the test choices.

The agent has the prior belief $f_0 = \big(f_0(\theta)\big)_{\theta \in \Theta}$ with $f_0(\theta) \in (0, 1)$ for $\forall \theta \in \Theta$. At the start of time t, the agent has the belief f_t. After choosing the action r_t, the agent receives the signal s_t and updates the belief to f_{t+1} using Bayes rule. I assume the agent has a uniform prior belief. That is $f_0(\theta) = \frac{1}{N}$ for $\forall \theta \in \Theta$.

At each time t, the agent either learns or fails to learn the true value of the parameter. If she learns the true value of the parameter at time t, then she gets a reward $x_t = x > 0$, and the game ends. If she fails to learn the true value of the parameter at time t, she gets $x_t = 0$ and the game enters time $t+1$. Let $u : \mathbb{R} \to \mathbb{R}_+ \cup \{0\}$ be the utility function that maps from the ex-post reward to the set of positive real numbers. The agent evaluates the ex-post reward x_t at time t by $u(x_t)$, where $u(\cdot)$ is increasing and $u(0) = 0$. The objective of the agent is to choose the strategy that maximises her lifetime utility

$$\mathbb{E}\left[\phi\left(\delta^{\tau} u(x)\right)\right], \tag{1}$$

where ϕ is strictly increases, τ is the time that the agent learns the value of the parameter, and the expectation is taken over the distribution of learning time τ. This lifetime utility is the generalised expected discounted utility introduced in Dejarnette et al. (2018). It allows for different time risk attitude. When $\phi(z) = z$, we have the standard exponential discounting.

2.2 Discussion

This model is a single agent's learning problem. The unknown parameter is predetermined and fixed overtime. The agent can costlessly do tests in order to obtain information about the unknown parameter. The results of the tests are purely informational. That is, the agent's flow payoff at time t does not depend on the results of the tests. Therefore, the agent does not have incentive to pass the test. These assumptions are similar to Meyer (1994), where the learner ('the principal' in their model) can costlessly design tasks to learn the type of the worker and the principal does not gain any payoffs related to the task completion. Deb & Stewart (2018) also has the similar learning feature, but the object the learner wants to learn in their model is a strategic player, and hence not a fixed parameter.

The agent in this paper just wants to learn the true value of the parameter. There is no exploitation and exploration tradeoffs. Since the agent only gets a reward when she learns the unknown parameter, there is no exploitation. The agent does not settle before learning the true value of the parameter. Therefore, the model is not about when the agent stop learning. It is about how the agent acquires information to learn the true parameter when she has different intertemporal preferences.

The parameter set is assumed to be finite and sorted. It is assumed to be finite so that the agent can eventually learn the true parameter. This assumption is equivalent to assuming the unknown parameter is drawn from a compact set, and the agent learns the unknown parameter when she is ϵ-close to the true value where ϵ is exogenous. The parameter set is assumed to be sorted to simplify the expression of the agent's action and the signal structure. If this assumption is dropped, the following specification of the agent's action and the signal structure is equivalent to what is in the model setup. The agent's action is to partition the parameter set into two subsets. The signal tells the agent which subset the true parameter belongs to. Then Linear Search and Binary Search should also be redefined. The partition is considered to be equivalent to Linear Search if one of the subsets only contains one element. The partition is considered to be equivalent to Binary

Search if the two subsets have the same number of elements. In the computer science literature, Binary Search always requires the parameter set to be sorted. But for the purpose of this paper, this assumption is not needed. This assumption is made to simplify the expression of the agent's action and the signal structure.

Time is finite and the duration of the game is long enough such that the agent always has enough time to learn the unknown parameter. [3] Linear Search requires the longest time to learn. If the cardinality of the parameter set is $\bar{N}$, it requires at most $\bar{N}-1$ periods to learn the unknown parameter. Therefore, with the assumption that $T \geq \bar{N}-2$, the agent always has enough time to learn.

The agent is assumed to be Bayesian. However, because of the signal structure, where the the signal is binary and noise free, the Bayesian assumption is not as demanding as in the literature. The agent just needs to be able to partition the parameter set and then re-scale the prior belief after receiving the signal. Bayes rule is not needed for the agent to revise her belief.

Finally, the agent's prior belief is assumed to be uniform. Uniform distribution is symmetric. It describes that all the parameters in the parameter set are equally likely. Following the Principle of Insufficient Reason, if there are N indistinguishable parameters, each of them should be assigned a probability $\frac{1}{N}$. In Bayesian probability, the uniform prior is the simplest diffuse prior. It basically describes that the agent only has vague information about the unknown parameter. The important property I exploit in this paper is the symmetry of the uniform prior. The result that the agent does not switch between Binary Search and Linear Search relies on this symmetry property. In Appendix A.2, I discuss a simple relaxation of the uniform prior assumption.

3 The benchmark case

I consider the case that $\phi(z) = z$ and $u(z) = z$ as the benchmark case. The agent discounts the future payoff exponentially, and the time-t utility is risk neutral. The lifetime utility now becomes

$$\max \mathbb{E}\left[\delta^{\tau} x\right], \tag{2}$$

where $\delta \in (0,1)$ is the discount factor. I normalise the reward x to 1 in the following discussion to simplify the calculation.

I first rewrite the agent's problem as a dynamic programming problem and then characterise the optimal strategy. The main results Proposition 1 characterise the optimal strategy.

[3] This requires that the agent does not choose fully uninformative signals. An alternative situation is when there exists a deadline such that the agent may not have enough time to learn. That is, the final period T is smaller than $\bar{N}-2$. If there were a deadline, it may result in different optimal search behaviour. An extreme case is that if the agent only has one period to learn the value of the parameter, then Linear Search is always optimal, as it allows the agent to learn with positive probability, while other search strategies do not. When a deadline exists, the agent might want to maximise the probability of learning the unknown parameter. The optimal search behaviours might be different from the optimal ones in this paper.

3.1 Dynamic programming setup

I first define the state variable and the choice at time t, and then formally define Binary Search and Linear Search. Then I write down the Bellman equation.

At each time t, by choosing the test r_t, the agent partitions the parameter set into two subsets. The signals tell the agent which set the true parameter belongs to. The agent then can eliminate the other subset. Therefore, from the agent's point of view, the parameter set shrinks to one of the subsets after receiving the signal. Since the agent remembers the past actions and the past signals, the optimal action at next time must belong to the subset that is not eliminated. It is then sufficient to keep track of the evolution of the parameter set. Let $\Theta_0 = \Theta$ be the initial parameter set. Let Θ_t be the parameter set at the beginning of time t. Since the agent's action can be considered as partitioning the parameter set, and the agent has uniform prior, the cardinality of the parameter set can be modelled as the state variable at time t. The action of the agent at time t is to choose how to partition the parameter set into two subsets. Let N_t be the cardinality of the parameter set Θ_t. Let m_t and n_t be the cardinalities of the two subsets the agent chooses. Thus, at time t, the state variable is N_t, and the choice is $(m_t, n_t) \in \mathcal{F}_t = \{(m_t, n_t)|m_t+n_t = N_t, (m_t, n_t) \in ((\mathbb{Z}_+)^2 \cap [1, N_t])^2\}$, where $\mathcal{F}_t$ is the feasible set of the choice at time t. Due to symmetry, assume without loss of generality that $m_t \leq n_t$. The evolution of the state is as follows. Given the state at time t and the choice (m_t, n_t), the state at time $t+1$ is

$$N_{t+1} = \begin{cases} m_t & \text{with probability } \frac{m_t}{N_t} \\ n_t & \text{with probability } \frac{n_t}{N_t} \end{cases}.$$

Next, I formally define *Binary Search*, *Linear Search* and *the agent learns the true parameter at time t* using the terminologies defined above.

Definition 1. The *Binary Search Policy in state* N_t is the choice $(m_t, n_t) \in \mathcal{F}_t$ such that $m_t = \lfloor \frac{N_t}{2} \rfloor$ and $n_t = \lceil \frac{N_t}{2} \rceil$ [4] . The *Binary Search Strategy* is the strategy that prescribes the agent the Binary Search Policy in all the states.

Definition 2. The *Linear Search Policy in state* N_t is the choice $(m_t, n_t) \in \mathcal{F}_t$ such that $m_t = 1$ and $n_t = N_t - 1$. The *Linear Search Strategy* is the strategy that prescribes the agent the Linear Search Policy in all the states.

Definition 3. The agent learns the true parameter at time t if $N_{t+1} = 1$ (N_{t+1} is the state at the end of the period t).

To simplify the expression, I ignore the t subscript in the following discussion. If the agent uses the Linear Search Policy in state N, the agent can learn the true parameter with probability $\frac{1}{N}$. The state evolves to $N-1$ in the next period with probability $\frac{N-1}{N}$. If the agent uses the choices other than the Linear Search Policy in state N, the agent cannot learn the true parameter today if $N > 2$, and the state evolves according to his choice. Define another set $\mathcal{F}^\dagger = \mathcal{F} \setminus \{(m = 1, n = N-1)\}$

[4]The notation $\lfloor x \rfloor$ rounds $x \in \mathbb{R}$ to the nearest integer less than or equal to x, and the notation $\lceil x \rceil$ rounds $x \in \mathbb{R}$ to the nearest integer greater than or equal to x.

to be the set of the choices excluding Linear Search in state N. The Bellman equation is

$$V(N) = \max\left\{\frac{1}{N} + \frac{N-1}{N}\delta V(N-1), \max_{(m,n)\in\mathcal{F}^\dagger} \delta\Big\{\frac{m}{N}V(m) + \frac{n}{N}V(n)\Big\}\right\}.$$

The first term is the value of the Linear Search Policy in state N and the second term is the value of other choices in state N. If $N = 2$, the agent learns the value of the parameter at the end of this period. That is, $V(2) = 1$. From this, the initial condition of the dynamic programming problem is $V(1) = \frac{1}{\delta}$. If there is only on element in the parameter set, it means that the agent learns the true parameter in the last time period. The Bellman equation can be simplified to

$$V(N) = \max_{(m,n)\in\mathcal{F}} \delta\Big\{\frac{m}{N}V(m) + \frac{n}{N}V(n)\Big\} \tag{3}$$

with the initial condition $V(1) = \frac{1}{\delta}$. The closed form of the value function when $N \leq 3$ are easy to compute, where $V(2) = 1$ and $V(3) = \frac{1}{3} + \frac{2}{3}\delta$. Let $W(N) := NV(N)$ be the product of N and the value function $V(N)$. The Bellman equation (3) can be rewritten as

$$W(N) = \max_{(m,n)\in\mathcal{F}} \delta\Big\{W(m) + W(n)\Big\}.$$

3.2 The optimal strategy

In this section, I will show that the Linear Search Strategy is optimal if the agent is sufficiently impatient, and the Binary Search Strategy is optimal otherwise. All other strategies are weakly sub-optimal.

Using the Linear Search Strategy allows the agent to test one element at each time in any state $N > 2$. The best-case scenario is that the agent learns the unknown parameter immediately at $t = 0$, but most likely the agent learns the unknown parameter at some other time $0 < t < \bar{N} - 2$. By using the Linear Search Strategy, the agent effectively check one element in the parameter set at a time. But, when there are only two parameters in the parameter set, learning that one parameter is not the true value of parameter is equivalent to learning that the other parameter left is. It is as if the agent is able to check both parameters at once. Since the agent's prior belief is uniform, the expected value associated with the Linear Search Strategy is thus the sum of the geometric series $\mathcal{L} = (\frac{1}{N}, \frac{1}{N}\delta, \ldots, \frac{1}{N}\delta^{N-3})$ plus $\frac{2}{N}\delta^{N-2}$.

Using the Binary Search Strategy, the agent can learn the value of the parameter within two consecutive periods. When the state is a power of two, say $N = 2^K$, where K is a positive integer, the agent learns the unknown parameter at time $\tau_2^N = K - 1$ with probability one. If the state is greater than 2^K and smaller than 2^{K+1}, the agent learns the unknown parameter at time $\tau_1^N = K$ or $\tau_2^N = K - 1$ with positive probabilities $\frac{\pi^N}{N}$ and $1 - \frac{\pi^N}{N}$ respectively. The uncertainties associated with the timing of learning is small compared to that of the Linear Search Strategy. The agent does face uncertainties in each state that is not a power of two. In an odd state, the uncertainty arises because the following state is not deterministic. In an even state, the uncertainty can still arise if the following state is odd. When starting from a large initial state $\bar{N}$, a big number of states can be visited before learning the unknown parameter. Since there can be uncertainties in each of the odd state visited, one may think that there could be a high aggregate uncertainty associated with the Binary Search Strategy. However, this is not true. This is because the agent only learns the unknown parameter when the state evolves to one. By using the Binary Search Strategy, in any state $N > 3$, the state never evolves to one directly. Instead, state one only occurs after states $N = 2$ and $N = 3$. When the state $N = 3$ occurs, with probability of a third, the agent learns the unknown parameter. When the state $N = 2$ occurs, the agent learns the parameter for sure today. If the state evolves to $N = 2$ directly without reaching state three, the agent learns the unknown for certain. If the state evolves to $N = 3$ before evolving to $N = 2$, the agent might learn the unknown parameter today or tomorrow. By using the Binary Search Strategy, the learning procedure ends in visiting either state $N = 2$ or $N = 3$. As a consequence, the agent always learn the unknown parameter within two consecutive periods. g

Let $V^L(\cdot)$ and $V^B(\cdot)$ be the values associated with the Linear Search Strategy and the Binary Search Strategy. Let $\pi^N = 2N - 2^{\lfloor \log_2 N \rfloor + 1}$, $\tau_1^N = \lceil \log_2 N \rceil - 1$, and $\tau_2^N = \lfloor \log_2 N \rfloor - 1$, where τ_1^N and τ_2^N are the two consecutive time at which the agent learns the unknown parameter, and $\frac{\pi^N}{N} \in [0, 1)$ is the probability that the agent learns the unknown parameter at time τ_1^N.

Lemma 1. *The value associated with the Linear Search Strategy is*

$$V^L(N) = \frac{1}{N}\left(\frac{1 - \delta^{N-1}}{1 - \delta} + \delta^{N-2}\right).$$

The value associated with the Linear Search Strategy is

$$V^B(N) = \frac{1}{N}\left[\pi^N \delta^{\tau_1^N} + (N - \pi^N)\delta^{\tau_2^N}\right].$$

Given the reward x, the function $V^L(\cdot)$ and $V^B(\cdot)$ decrease in N. Let $W^L(N) := NV^L(N)$ and $W^B(N) := NV^B(N)$.

Lemma 2. *When $\delta > 0.5$ ($\delta < 0.5$), the first-order difference of $W^L(\cdot)$ is positive (negative), and the second-order difference of $W^L(\cdot)$ is negative (positive). The first-order difference of $W^B(\cdot)$ is positive (negative), and the second-order difference of $W^B(\cdot)$ is non-positive (non-negative). When $\delta = 0.5$, both $W^L(N)$ and $W^B(N)$ are independent of N.*

Lemma 1 and Lemma 2 can be used to show the optimality of the Linear and Binary Search Strategy.

Proposition 1. *There exists a unique threshold $\bar{\delta} = 0.5$ such that the Linear Search Strategy is weakly optimal if the agent has a discount factor $\delta \leq \bar{\delta}$, and the Binary Search Strategy is weakly optimal if the agent has a discount factor $\delta \geq \bar{\delta}$.*

To check the optimality of the Linear (Binary) Search Strategy, I define a *Linear (Binary) Search Deviating Strategy* and then show that the *Linear (Binary) Search Deviating Strategy* that gives the agent a higher payoff than the Linear (Binary) Search Strategy does not exist. A *Linear (Binary) Search Deviating Strategy* is a one-step deviation strategy from the Linear Search Strategy, such that the agent follows Linear (Binary) Search Strategy in all the states $n \neq N$, and deviates from the Linear (Binary) Search Policy in state N.

When the agent deviates from the Linear Search Policy to some other policy $(m, n) \in \mathcal{F}^{\dagger}$ in state N, the payoff today will be zero. The most profitable one-step deviation strategy hence must maximise the continuation value. If the agent uses the $(m, n) \in \mathcal{F}^{\dagger}$ policy in state N, then, the (undiscounted) continuation value is the convex combination of $V^L(m)$ and $V^L(n)$, with the weights being $\frac{m}{N}$ and $\frac{n}{N}$. Since $m \leq n$ and $V^L(\cdot)$ is decreasing, the value of $V^L(m)$ is greater than the value of $V^L(n)$, but the weight attaches to $V^L(m)$ is smaller than the weight attaches to $V^L(n)$. Increasing m (i.e. decreasing n) decreases the value of $V^L(m)$, but puts a higher weight to that value. At the same time, it increases the value of $V^L(n)$, but decreases the weight attaches to $V^L(n)$. As a consequence, it is ambiguous whether increasing m and decreasing n increases the continuation value. Whether increasing m is optimal depends on the value of the function $W^L(\cdot)$. According to Lemma 2, the function $W^L(\cdot)$ is decreasing and convex when $\delta \leq \bar{\delta}$. Therefore, the increase of the continuation value from decreasing n is smaller than the decrease of the continuation value from increasing m. Thus, if the agent were to deviate from the Linear Search Strategy in state N, the most profitable Linear Search Deviating Strategy is to choose $(m, n) = (2, N-2)$ in state N.

In state N, When the agent deviates from the Linear Search Policy in state N to $(m, n) = (2, N-2)$, the agent gives up the flow payoff of $\frac{1}{N}$ today, and increases the (discounted) continuation value from $\delta \frac{N-1}{N} V^L(N-1)$ to $\delta[\frac{N-2}{N} V^L(N-2) + \frac{2}{N} V^L(2)]$. To check whether the one-step deviation is optimal, the agent compares the value she gives up today (the loss) with the increase of the discounted continuation value (the gain). If the loss and the gain are scaled up by N, then the rescaled loss is a constant and the rescaled gain is decreasing in N when $\delta < \bar{\delta}$. That is, the rescaled gain is maximised in the smallest state $N = 4$. Therefore, if it is not optimal for the agent to deviate from the Linear Search Policy in state $N = 4$, then, the agent would not want to deviate from the Linear Search Policy in any other state that is greater than four. In state $N = 4$, when $\delta \leq \bar{\delta}$, the discounted increase in the continuation payoff by deviating from the Linear Search Policy is smaller than the payoff given up today. The agent hence does not want to deviate in state $N = 4$, and does not want to deviate in any other states greater than four.

When the agent deviates from the Binary Search Strategy to the policy $(m, n) \in \mathcal{F}^{\dagger}$ in state N, it does not change the payoff today, which is still zero. but changes the continuation values. If the continuation value in state N is scaled up by N, then, given any policy $(m, n) \in \mathcal{F}^{\dagger}$ in state N, the rescaled continuation value is the sum of the two elements $mV^B(m)$ and $nV^B(n)$. By definition, the Binary Search Policy in state N maximises the value of m. Whether decreasing m increases the continuation value depends on the property of the function $W^B(\cdot)$. According to Lemma 2, $W^B(\cdot)$ function is increasing and concave when $\delta > \bar{\delta}$. Since m is assumed to be smaller than n, decreasing m by one does not offset the effect of increasing n by one. Therefore, deviating to the policy $(m, n) \in \mathcal{F}^{\dagger}$ in state N is not optimal when δ is sufficiently big.

If instead of deviating to a policy $(m, n) \in \mathcal{F}^{\dagger}$, the agent deviates to the Linear Search Policy in state N, effectively, the agent postpones Binary Search until tomorrow and takes the risk of learning the unknown parameter today. If the agent fails to learn the unknown parameter today, she gets a small benefit of decreasing the state by one in next period. The cost and benefit of deviating to the choice Linear Search in state N are different in different states. It is more beneficial to deviate to the Linear Search Policy in a small state, say in state $N = 4$, rather than in a big state, say in state $N = 400$. First, it is because the probability of learning the unknown parameter, which is $\frac{1}{N}$, is higher when N is small. Second, it is because Binary Search is more efficient in terms of eliminating the number of impossible candidates when the state is big. For example, in state $N = 400$, the Binary Search Policy today will eliminate 200 impossible candidates, while in state $N = 4$, the Binary Search Policy will only eliminate 2 impossible candidates. Because of the two reasons above, if the agent would like to deviate to the Linear Search Policy in some state, it is most profitable for him to deviate in state $N = 4$, rather than in state $N = 400$. If deviating to the Linear Search Policy in state $N = 4$ is not optimal, then deviating to Linear Search Policy in other states is also not optimal. The gain from deviating to the Linear Search Policy in state $N = 4$ is $\frac{1}{4}x$, and the loss is $V^B(4) - \delta\frac{3}{4}V^(3)$. When $\delta > \bar{\delta}$, the gain is always smaller than the loss. Deviating to the Linear Search Policy in state $N = 4$ is thus not optimal. Therefore, when $\delta > \bar{\delta}$, deviating to the Linear Search Policy in any state N is not optimal.

3.3 Discussion

The thresholds $\bar{\delta}$ in Proposition 1 is a constant $\frac{1}{2}$. This means that for any value $\delta \in (0, 1)$, either the Binary Search Strategy or the Linear Search Strategy are *weakly* optimal. Some of the alternative strategies are weakly sub-optimal but not *strictly* sub-optimal because they induce the same distribution of the learning time as is induced by the Linear or Binary Search Strategy. For example, when $N = 6$, the following two strategies induce the same distribution of the learning time. That is, the agent learns the parameter at $t = 1$ with probability $\frac{1}{3}$ and learns the parameter at $t = 2$ with probability $\frac{2}{3}$.

Strategy One: The Binary Search Strategy.

Strategy Two: The agent chooses $(m, n) = (2, 4)$ *in state* $N = 6$*. If the state* 4 *occurs, the agent chooses* $(m, n) = (2, 2)$*. and if the state* 2 *occurs, the agent chooses* $(m, n) = (1, 1)$*.*

Since these two strategies induce the same distribution of the learning time, these two strategies induce the same expected utility. The *Strategy Two* above is only weakly dominated by the Binary Search Strategy.

The threshold $\bar{\delta}$ is a constant, and is independent of the state N. For example, if an agent has the discount parameter $\delta = 0.45$, then it is optimal for him to use the Linear Search Policy in state $N = 4$, and it is also optimal for the agent to use the Linear Search Policy in state $N = 400$. If the agent has the discount parameter $\delta = 0.9$, then it is optimal for him to use the Binary Search Policy in state $N = 4$ and in state $N = 400$. Thus the agent does not need to commit to these policies, they are the consequences of the optimal choices at every state. The simple form of the policy is due to the simple form of the state variable. The state variable is just the cardinality of the parameter set because of the uniform prior assumption. Since the agent's prior belief is assumed to be uniform, the agent's posterior belief is still uniform. The agent always believes that the remaining candidates are equally likely, that is, the likelihood ratio of any two candidates is one. The number of candidates in the parameter set only scales up or down the absolute value of the probability attached to each candidate, but does not change the likelihood ratio of any two candidates. As a result, with the uniform prior assumption, in different states, the agent is facing the problem with the same properties: all the elements in the parameter set are equally likely to be the true value of the parameter. If the likelihood ratio of any two elements changes as the state changes, the agent might use different policies in different states. For example, let N be the cardinality of the initial parameter set, which is the initial state. Consider the agent's prior belief to be such that the first element θ_1 in the parameter set is the most likely one with probability a half, and all other parameters are equally likely. The likelihood ratio of θ_1 to other elements is $N-1$. In this case, since θ_1 is very likely compared to other elements in the parameter set, the agent has a strong incentive to choose the Linear Search Policy in state N regardless the value of δ. If the agent does not learn the unknown parameter, the state $N-1$ occurs and the agent's belief is revised to a uniform distribution. In state $N-1$, the agent hence is facing the problem as in the benchmark case. The likelihood ratio of any two elements is one. The agent's strategy then depends on δ. In Appendix A.2, I consider another prior belief to check how the it can affect the agent's optimal strategy.

The argument above appears to say that being patient or impatient explains why the Linear Search Strategy or Binary Search Strategy is optimal. But, there are other models of time preference where this apparent relationship fails. Consider the expected utility with a linear discounting, that is, the agent pays a fixed cost c at each time t if she searches and does not learn the unknown parameter (see Appendix A.1 for details). My main result in this case is that the Binary Search Strategy is *always* optimal as long as the cost c is positive and bounded. The agent is also impatient in this case, but this impatience is modelled by a linear discounting. Even though the linear discounting and the exponential discounting both describe the agent being impatient, the optimal strategies in these case are different from each other. In fact, the expected utility with the exponential discounting does not just describe the agent being impatient, it also describes the agent being "time risk loving". To understand *time risk*, consider the following two options one can choose from. These are just illustrative examples to show what time risk refers to [5].

Option One: Learn the unknown parameter today ($t = 0$) or the day after tomorrow ($t = 2$) with equal probabilities.

Option Two: Learn the unknown parameter tomorrow ($t = 1$) for certain.

If the agent discounts future utilities exponentially, Option One gives the agent an ex-ante utility of $\frac{1}{2}(1+\delta^2)x$, and Option Two gives the agent an ex-ante utility of δx. The agent then *always* prefers Option One, as it gives the agent a higher expected utility with any $\delta < 1$. Since these two options yield the same expected learning time: tomorrow ($t = 1$), and the agent prefers the one that is risky in terms of learning time, the agent is hence time risk loving. If the agent discounts future utilities with linear discounting, then the agent is indifferent between these two. The agent is hence time risk neutral.

The comparison between the two different preferences shows that the time risk preference affects the optimal strategy. In fact, the fixed search cost utility can be expressed as the utility function Equation (1) with $\phi(z) = \log(z)$. The $\phi(\cdot)$ function determines the agent's time risk attitude (Dejarnette et al. (2018)) and affects the agent's optimal strategy.

[5] This example and the discussion of the time risk are borrowed from Dejarnette et al. (2018).

4 Time risk attitude

In this section, I consider the lifetime utility as in eq. (1). The lifetime utility is identified with a triple (ϕ, δ, u). As discussed in Dejarnette et al. (2018), the agent's time risk attitude is determined by the concavity of ϕ, the intertemporal substitution is determined by δ and ϕ, and the atemporal risk attitude towards the lotteries with only immediate payment is determined by $\phi \circ u$. I show that if ϕ is a power function with power $1-\gamma$ and u is the inverse of ϕ, then, the optimal strategy of the agent with utility (ϕ, δ, u) is the same as the benchmark agent (the agent with utility function as in eq. (2)) with a discount parameter ρ where $\rho = \delta^{1-\gamma}$.

If ϕ is a power function, say $\phi(z) = \frac{1}{1-\gamma} z^{1-\gamma}$ where $\gamma \in (0,1) \cup (1, +\infty)$, then, eq. (1) can describe both time risk loving and time risk averse utility. The following lemma presents this result.

Lemma 3. *If $\phi(z) = sign(1-\gamma) a z^{1-\gamma}$ where $a > 0$ is a constant, then, for any increasing u function, when $\gamma \in (0,1)$, (ϕ, δ, u) describes time risk loving preference, and when $\gamma > 1$, (ϕ, δ, u) describes time risk averse preference.*

This lemma follows directly from Dejarnette et al. (2018) Prop 2. To understand why this is true, reconsider the two options introduced in Section 3.

Option One: Learn the unknown parameter and get a reward of one today ($t = 0$) or the day after tomorrow ($t = 2$) with equal probabilities.

Option Two: Learn the unknown parameter and get a reward of one tomorrow ($t = 1$) for certain.

Suppose $\phi(z) = \frac{1}{1-\gamma} z^{1-\gamma}$. Given the preference (ϕ, δ, u), the utility from Option One is $\frac{1}{1-\gamma}(\frac{1}{2}u(1)^{1-\gamma} + \frac{1}{2}[\delta^2 u(1)]^{1-\gamma})$, and the utility from Option Two is $\frac{1}{1-\gamma}[\delta u(1)]^{1-\gamma}$. When $\gamma \in (0,1)$, the utility from Option One is greater than the utility from Option Two. When $\gamma > 1$, the utility from Option One is smaller than the utility from Option Two, and The special cases discussed in Section 3, the exponential discounting and the search with a fixed cost, are limiting cases. When $\gamma \to 0$, $\phi(z)$ converges to $\phi(z) = z$, which is the exponential discounting case. When $\gamma \to 1$, $\phi(z)$ converges to the natural log function, which describes the linear discounting preference.

Next, I show that there exists a (ϕ, δ, u) such that given any strategy, the utility derived from (ϕ, δ, u) equals the utility derived from the exponential discounting with a different discount parameter ρ. In addition, such functional forms of ϕ and u are unique.

Lemma 4. *Let $a \neq 0$ be a constant. Then, $\phi\big(\delta^\tau u(x)\big) = \rho^\tau x$ if and only if $\hat{\phi}(z) = a z^{\frac{\log \rho}{\log \delta}}$ and $\hat{u}(x) = \big(\frac{x}{a}\big)^{\frac{\log \delta}{\log \rho}}$. If in addition $\rho = \delta^c$ where $c \neq 0$, then, $\hat{\phi}(z) = a z^c$ and $\hat{u}(x) = \big(\frac{x}{a}\big)^{\frac{1}{c}}$.*

Each strategy pins down a distribution of the learning time. According to Lemma 4, given a strategy, the utility derived from $(\hat{\phi}, \delta, \hat{u})$, which is $\mathbb{E}\big[\phi\big(\delta^\tau u(x)\big)\big]$, equals the utility derived from the exponential discounting with a different discount parameter ρ, which is $\mathbb{E}\big[\rho^\tau x\big]$. This shows that the agent with preference $(\hat{\phi}, \delta, \hat{u})$ behaves as if she discounts future payoffs exponentially with a discount parameter $\rho = \delta^c$. Because of this equivalence of the behaviour, this lemma together with Proposition 1 can be used to find the optimal strategy of the agent with preference $(\hat{\phi}, \delta, \hat{u})$.

In the following discussion, let $\hat{\phi}(z) = \frac{1}{1-\gamma} z^{1-\gamma}$ and $\hat{u}(x) = [(1-\gamma)x]^{\frac{1}{1-\gamma}}$ where $\gamma \in (0,1) \cup (1,\infty)$. This specification is not necessary for the following results, but it gives a clearer intuition. I refer to $\rho = \delta^{1-\gamma}$ as a *virtual discount parameter.*

The following proposition characterises the optimal strategy for preference $(\hat{\phi}, \rho, \hat{u})$.

Proposition 2. *For each $\gamma \in (0,1)$, there exists a unique threshold $\breve{\delta}_\gamma := (\bar{\delta})^{\frac{1}{1-\gamma}} < \bar{\delta} = \frac{1}{2}$, such that the Linear Search Strategy is optimal if $\delta \leq \breve{\delta}_\gamma$ and the Binary Search Strategy is optimal if $\delta \geq \breve{\delta}_\gamma$. The threshold $\breve{\delta}_\gamma$ is decreasing in γ, and it approaches $\bar{\delta}$ when γ approaches zero.*

For $\gamma > 1$, the Binary Search Strategy is optimal given any discount parameter $\delta \in (0,1)$.

When $\gamma \in (0,1)$, as shown in Lemma 4, the agent with preference $(\hat{\phi}, \delta, \hat{u})$ behaves as if she discounts the future payoffs exponential with a discount parameter $\rho = \delta^{1-\gamma}$. Given Proposition 1, the Linear Search Strategy is optimal if the discount parameter ρ is weakly smaller than a half. That is, the Linear Search Strategy is optimal if δ is weakly smaller than $\left(\frac{1}{2}\right)^{\frac{1}{1-\gamma}} := \breve{\delta}_\gamma$. The similar argument holds for the Binary Search Strategy. Since $\gamma \in (0,1)$, $\breve{\delta}_\gamma$ decreases in γ.

When $\gamma \in (0,1)$, the agent's virtual discount parameter ρ is greater than his real discount parameter δ. Due to the effect of the time risk aversion, the agent behaves as if she is more patient. For example, consider an agent A who has the benchmark utility and discounts future payoffs exponentially with the discount parameter $\tilde{\delta} < \frac{1}{2}$. The optimal strategy of this agent is then the Linear Search Strategy. Consider another agent B with preference $(\hat{\phi}, \tilde{\delta}, \hat{u})$ where $\gamma \in (0,1)$. Suppose γ takes the value such that the optimal strategy of the agent B is the Binary Search Strategy. Imagine there is an outside observer observing these two agents' strategies. If the outside observer incorrectly believes that the agent B has the preference as in the benchmark case, then the outside observer would conclude that the agent B is more patient than the agent A. However, the fact is that the agent B is as patient as the agent A, but less time risk loving. The decreasing $\breve{\delta}_\gamma$ shows that when γ increases, the agent becomes less time risk loving, and has a stronger incentive to use the Binary Search Strategy. Being less time risk loving, the less patient agent still prefers to use the Binary Search Strategy.

When $\gamma > 1$, the agent is time risk averse. Since we know from Section 3 that the Binary Search Strategy is optimal for the time risk neutral agent as it induces a distribution of learning time with a smaller mean, when the agent becomes time risk averse, the agent prefers the Binary Search Strategy more. Since the Binary Search Strategy induces a distribution of the learning time with a smaller variance, it is hence also optimal for the risk averse agent.

Figure 2 demonstrates the optimal strategy given the value of the discount parameter δ and the time risk aversion parameter γ. From Proposition 2, we know that the functional form of the red curve in Figure 2 is $\delta^{1-\gamma} = \frac{1}{2}$. When the discount parameter δ is greater than a half, the optimal strategy is the Binary Search Strategy given any value of the time risk aversion parameter γ. That is, when the agent is sufficiently patient, the effect of being patient dominates the effect of time risk attitude. It is optimal for the agent with $\delta > \frac{1}{2}$ to use the Binary Search Strategy regardless the time risk attitude. When the time risk aversion parameter γ is greater than one, the optimal strategy is the Binary Search Strategy given any value of δ. That is, when the agent is time risk averse, the effect of the time risk attitude dominates the effect of patience. It is optimal for the time risk averse agent to use the Binary Search Strategy regardless the patience level. When the agent is time risk

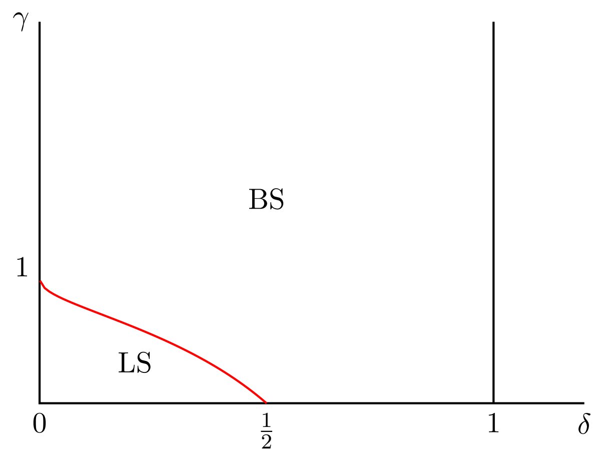


Figure 2: The optimal strategy for each value of the discount parameter δ and the risk-aversion parameter γ. The functional form of the red curve is $\delta^{1-\gamma} = \frac{1}{2}$.

loving and not sufficiently patient, that is, when $\delta < \frac{1}{2}$ and $\gamma < 1$, there is a trade-off between learning in a risky way and learning faster in expectation. Being time risk loving incentivises the agent to take the risk of learning the parameter today, while being patient incentivises the agent to learn faster in expectation. As a consequence, the optimal strategy depends on the effect that dominates.

Given the value of γ, the threshold $\breve{\delta}_\gamma$ in Proposition 2 is a constant, and it does not depend on N. This means that the agent's optimal strategies have the same time-consistent property as in Section 3. That is, if the agent has the discount parameter δ and the risk-aversion parameter γ such that $\rho = \delta^{1-\gamma} = 0.45$, then it is optimal for him to use the Linear Search Policy in state $N = 4$ and the state $N = 400$. If the agent has the discount parameter δ and the risk-aversion parameter γ such that $\rho = \delta^{1-\gamma} = 0.9$, then it is optimal for him to use the Binary Search Policy in state $N = 4$ and the state $N = 400$. Therefore, the agent does not need to commit to a certain policy, the commitment to a certain policy along the path of learning is a result of the optimal strategy. This is true because of the equivalence condition specified in Lemma 4. The optimal strategies in this section thus preserve the same properties as in the benchmark case.

The Linear Search Strategy is not only risky in terms of the timing of learning the unknown parameter, it is also risky in terms of whether the agent can learn at a certain time t. That is, if N is the state at time t, by using the Linear Search Strategy, the agent learns the parameter with the probability $\frac{1}{N}$, and does not learn the parameter with the probability $\frac{N-1}{N}$. There are two different types of risks associated with the Linear Search Strategy. One is the time risk that has been discussed in the previous part of this section, and the other one is the temporal risk at each time t. Because $\hat{u}$ is the inverse of $\hat{\phi}$, the agent is assumed to be risk-neutral towards time-t lotteries. To discuss the effect of the temporal risk attitude, I consider EZ preferences in Section 5.

5 Epstein and Zin preferences

In order to discuss the EZ preferences, I first rewrite the one time reward at time t as a stream of reward over time. Since the agent essentially gets a reward of zero at the time when the parameter is not learned, learning the parameter at time τ is equivalent to getting a reward stream $(x_1, x_2, \ldots, x_\tau, x_{\tau+1} \ldots, x_T) = (0, 0, \ldots, x, 0, \ldots, 0)$. After introducing this reward stream, the EZ recursive setup can be used to evaluate the agent's lifetime utility.

The recursive formulation of EZ preferences is developed in Epstein & Zin (1989), which is originated in Kreps & Porteus (1978) in a finite time setting. The EZ recursive utility function consists of two components: a CES time aggregator that characterises the preference over the deterministic payoff vector, and a risk aggregator that aggregates the risk associated with future uncertain payoffs. Consider a deterministic payoff vector $(z_0, z_1, \ldots, z_T)$, with z_t denoting the payoff at time t. The utility from time t onwards is

$$U(z_t, z_{t+1}, \ldots, z_T) = \left(z_t^\rho + \delta U(z_{t+1}, \ldots, z_T)^\rho\right)^{\frac{1}{\rho}} \tag{4}$$

where $\delta \in (0, 1)$ is the discount factor and $\frac{1}{1-\rho} > 0$ is the elasticity of intertemporal substitution (EIS). The greater the value of EIS, the greater willingness the agent has (or, the easier it is) to substitute today's payoff to tomorrow's payoff. In case of the uncertain payoffs, the utility is evaluated by a certainty equivalent operator that is introduced in Kreps & Porteus (1978). Let z be a set of future payoff vectors. Let p be a probability measure on the set z. Then the utility from the uncertain payoffs is

$$\left(\mathbb{E}_p U(z)^\alpha\right)^{\frac{1}{\alpha}} \tag{5}$$

where $1 - \alpha > 0$ is the relative risk aversion (RRA). A smaller value of α corresponds to a greater risk aversion.

How does this utility specification affect the agent's evaluation of Binary Search and Linear Search? The Binary Search Policy ensures the agent a certain payoff today, either zero or one, but the Linear Search Policy gives the agent an uncertain payoff both today and in the future. To be more specific, if the agent chooses the Binary Search Policy at time t, then the payoff at time t is deterministic and the future payoffs are uncertain. The agent then evaluates the uncertain future payoffs by the certainty equivalent operator, and the total utility at time t aggregates the certain payoff at time t and the certainty equivalent of the future uncertain payoffs using the CES aggregator. If the agent chooses the Linear Search Policy at time t, the payoff today and the payoffs in the future are both uncertain. To evaluate the total utility associated with the uncertain payoffs, the agent first evaluates the utility of each payoff stream using the CES aggregator. The total utility is then calculated using the certainty equivalent operator. In other words, when the agent chooses Binary Search, the CES aggregator is the 'outside operator'. The certainty equivalent operator is only used to evaluate the future payoffs, and hence is the 'inside operator' When the agent chooses the Linear Search Policy, the CES aggregator becomes the inside aggregator, and the certainty equivalent operator is the outside operator.

Given the recursive formulation of the EZ preference, the Bellman equation in state N is

$$V^{EZ}(N) = \max\Big\{\Big(\frac{1}{N}(1^\rho + \delta 0^\rho)^{\frac{\alpha}{\rho}} + \frac{N-1}{N}(0^\rho + \delta V^{EZ}(N-1)^\rho)^{\frac{\alpha}{\rho}}\Big)^{\frac{1}{\alpha}},$$
$$\max_{(m,n)\in\mathcal{F}^\dagger}\Big(0^\rho + \delta\big(\frac{m}{N}V^{EZ}(m)^\alpha + \frac{n}{N}V^{EZ}(n)^\alpha\big)^{\frac{\rho}{\alpha}}\Big)^{\frac{1}{\rho}}\Big\}$$

with the initial condition $V^{EZ}(1) = \delta^{\frac{\rho}{\alpha^2}}$. The initial condition is computed from the fact that $V^{EZ}(2) = 1$, that is, when the state $N = 2$, the agent learns the true value of the parameter immediately after one search. The first element in the Bellman equation is the value associated with the Linear Search Policy in state N. By choosing the Linear Search Policy today, the agent either gets a payoff of one today and zero tomorrow, or a payoff of zero today and the continuation value $V^{EZ}(N-1)$ tomorrow. It is then as if the agent is facing two possible payoff vectors: $(1, 0)$ and $(0, V^{EZ}(N-1))$, with probability $\frac{1}{N}$ and $\frac{N-1}{N}$ respectively. The utility is evaluated by the certainty equivalent of the utilities associated with the two payoff vectors. The second element in the Bellman equation is the value associated with the policy $(m, n) \in \mathcal{F}^\dagger$ in state N. By choosing the policy $(m, n) \in \mathcal{F}^\dagger$, the payoff of zero today is certain. The continuation value however, is $V^{EZ}(m)$ or $V^{EZ}(n)$ with probability $\frac{m}{N}$ and $\frac{n}{N}$. The certainty equivalent operator is used to calculate the value associated with the future payoff, and the CES aggregator aggregates the payoff today and the certainty equivalent of the payoff tomorrow. The agent chooses the policy (m, n) that gives the agent the highest value. With the initial condition $V^{EZ}(1) = \delta^{\frac{\rho}{\alpha^2}}$, the Bellman equation can be simplified to

$$V^{EZ}(N) = \max_{(m,n)\in\mathcal{F}}\Big(\zeta\Big(\frac{m}{N}V^{EZ}(m)^\alpha + \frac{n}{N}V^{EZ}(n)^\alpha\Big)\Big)^{\frac{1}{\alpha}}, \tag{6}$$

where $\zeta \equiv \delta^{\frac{\alpha}{\rho}}$ is a virtual discount parameter.

In general, the parameters α and ρ can take any value smaller than one, that is, both of the parameters can be negative. But, in the following discussion, I restrict the values of α and ρ to be positive. First, to make sure the parameter ζ is indeed a virtual discount parameter, I restrict the parameters α and ρ to have the same sign. Second, Epstein & Zin (1989) points out that decreasing α can change the agent's preference in terms of the timing of the resolution of uncertainty. To shut down this possible effect, I restrict the value of α to be positive. That is, I only consider the case that the agent is not 'too risk averse', and it is not 'too hard' to substitute the consumption intertemporally.

5.1 The optimal strategy

This section discusses the agent's optimal strategy and compares it with the benchmark case.

To find the agent's optimal strategy in this section, I show that the Bellman equation (6) is closely related to the Bellman equation (3) in the benchmark case. Let $\mathcal{W}(N) = V^{EZ}(N)^\alpha$.

Equation (6) can be rewritten as

$$\mathcal{W}(N) = \max_{(m,n)\in\mathcal{F}} \zeta\Big(\frac{m}{N}\mathcal{W}(m) + \frac{n}{N}\mathcal{W}(n)\Big) \tag{7}$$

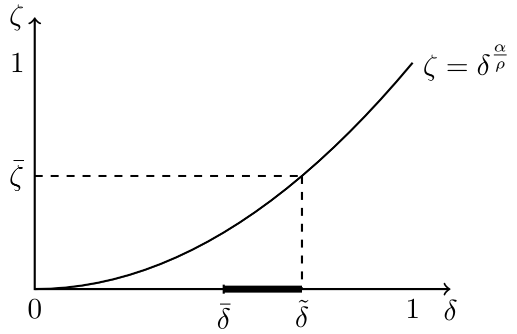
with the initial condition $\mathcal{W}(1) = \frac{1}{\zeta}$. Equation (7) is essentially the same Bellman equation as Equation (3) with the same initial condition. The only difference is that the discount parameter in Equation (7) is the virtual discount parameter ζ, and the discount parameter in Equation (3) is the real discount parameter δ. Since the Bellman equation in this section and the Bellman equation in the benchmark case are closely related in the way described above, the agent with EZ preferences behaves the same as the agent in the benchmark case with a discount parameter ζ. In the benchmark case, the value of the real discount parameter determines the agent's optimal strategy. In this section, the virtual discount parameter ζ hence determines the optimal strategy of the agent with EZ preferences. That is, if ζ is greater than a half, the Binary Search Strategy is optimal. If ζ is smaller than a half, the Linear Search Strategy is optimal. Since the result Proposition 1 in the benchmark case are presented as how the real discount parameter δ affects the agent's optimal strategy, I characterise the agent's optimal strategy below in the similar way. Let $\bar{\zeta} = \frac{1}{2}$.

Proposition 3. *If the agent has EZ preferences, given any (α, ρ) pair, there exists a unique threshold $\tilde{\delta} = (\bar{\zeta})^{\frac{\rho}{\alpha}}$ such that if $\delta > \tilde{\delta}$, the Binary Search Strategy is the optimal strategy. If $\delta < \tilde{\delta}$, the Linear Search Strategy is the optimal strategy. If $\delta = \tilde{\delta}$, the agent is indifferent between the Binary Search Strategy and the Linear Search Strategy.*

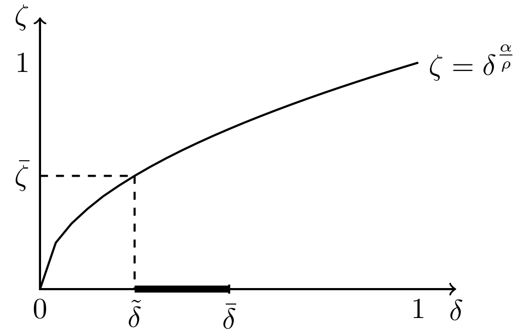
To better understand how the benchmark agent and the agent with EZ preferences behave differently, I consider the marginal agent who is just indifferent between the Linear Search Strategy and the Binary Search Strategy. The marginal agent in the benchmark case (the benchmark marginal agent) has a discount parameter $\delta = \bar{\delta}$, and the marginal agent with EZ preference (the EZ marginal agent) has the parameter triple (δ, α, ρ) such that $\delta = \tilde{\delta} = (\bar{\zeta})^{\frac{\rho}{\alpha}}$. Since the two thresholds $\bar{\delta}$ and $\bar{\zeta}$ are indeed the same, the relationship of $\bar{\delta}$ and $\tilde{\delta}$ depends on the value of the ratio of ρ and α. Proposition 3 hence can be used to check which marginal agent is more patient. Note that $EIS = \frac{1}{1-\rho}$ and $RRA = 1 - \alpha$.

Lemma 5. *If $\alpha > \rho$ ($EIS < \frac{1}{RRA}$), then $\tilde{\delta} > \bar{\delta}$. If $\alpha < \rho$ ($RRA > \frac{1}{EIS}$), then $\tilde{\delta} < \bar{\delta}$.*

Lemma 5 says that if EIS is smaller than the reciprocal of RRA, then the EZ marginal agent is more patient than the benchmark marginal agent. If RRA is greater than the reciprocal of EIS, then the EZ marginal agent is less patient than the benchmark marginal agent. This can be better understood by considering the agent's incentives and the features of the Linear and Binary Search Strategies. When EIS is small, it is hard to substitute the payoffs intertemporally. The agent hence has stronger incentives to use Linear Search Strategy, as it allows the agent to get immediately payoff. However, this payoff is uncertain. Therefore, there is a tradeoff between getting the payoff today and facing uncertain payoffs. The optimality of Linear Search or Binary Search depends on the value of EIS and RRA. If it is sufficiently hard to substitute intertemporal payoffs such that the gain of getting a payoff today overweighs the agent's aversion towards risky payoff, that is, if



(a) EIS dominates



(b) RRA dominates

Figure 3: The relationship between $\bar{\delta}$ and $\tilde{\delta}$

$EIS < \frac{1}{RRA}$, then the agent with EZ preference is willing to use Linear Search even when the agent has the discount parameter $\delta > \bar{\delta}$. This case is referred to the case that EIS dominates RRA. If, however, the agent is sufficiently risk averse such that the aversion towards risky overweighs the benefits of getting the payoff today, that is, if $RRA > \frac{1}{EIS}$, then the agent with EZ preference is willing to use Binary Search even when the agent has the discount parameter $\delta < \bar{\delta}$. This case is referred to as the case that RRA dominates EIS. Figure 3 plots the virtual discount parameter ζ as a function of the real discount parameter δ. The left panel plots the case that EIS dominates RRA, and the right panel plots the case that RRA dominates EIS. When EIS dominates RRA, it is optimal for the agent with the discount parameter $\delta \in (\bar{\delta}, \tilde{\delta})$ (thick black line on the left panel) to use the Binary Search Strategy in the benchmark case, but to use the Linear Search Strategy in the EZ preference case. This is because Linear Search satisfies the agent's preference on not willing to substitute intertemporal payoffs. When RRA dominates EIS, it is optimal for the agent with the discount parameter $\delta \in (\tilde{\delta}, \bar{\delta})$ (thick black line on the right panel) to use the Linear Search Strategy in the benchmark case, but to use the Binary Search Strategy in the EZ preference case. This is because Binary Search is less risky than Linear Search. Since the agent is very risk averse, she prefers Binary Search.

6 The present-biased preference

The optimal strategies in previous sections all have a consistent property. That is, if it is optimal for the agent to use the Linear Search Policy in state $N = 400$, then it is also optimal for the agent to use the Linear Search Policy in state $N = 4$. It is not optimal for the agent to switch between actions. In this section, I show that if the agent has a time-inconsistent preference, there exists the situation such that it is optimal for the agent to use the Binary Search Policy when the state is big, say $N = 400$, and it is optimal for him to use the Linear Search Policy when the state is small, say $N = 4$.

This section discusses the present-biased preference. Consider the case that an agent is asked which of the two options she prefers: one is to get a payment of one on Wednesday, and the other one is to get a payment of two on Friday. If the agent has the time-consistent preference, she will make the same decision regardless the question is asked on Monday or Wednesday. If the agent has the present-biased preference, she may, however, make different decisions when asked on different days. That is, when asked on *Monday* and *Tuesday*, the agent prefers to get the payment of two on Friday, but when asked on *Wednesday*, she changes his mind and prefers to get the payment of one on Wednesday, which is *today*. The agent changes his mind when the *future* becomes the *present*. The agent is biased towards the payment at the *present* when asked on Wednesday and thus has a stronger incentive to get the payment today.

These present-biased preferences could be described by quasi-hyperbolic discounting (see Phelps & Pollak (1968), Laibson (1997), Fischer (1999) and O'Donoghue & Rabin (1999)). To discuss the agent's present-biased preference, I consider the reward stream as in Section 5. Since the agent's preference changes over time, it is as if the agent has different selves at different times. The agent's time-t self evaluates the payments at and after time t as

$$\mathbb{E}\left[x_t + \beta \sum_{\tau=t+1}^{T} \delta^{\tau-t} x_\tau\right], \tag{8}$$

where $\beta \in (0,1)$ is the present-biased parameter, and $\delta \in (0,1)$ is the time-consistent discount parameter. The present-biased parameter β describes the magnitude of the present bias. When $\beta = 1$, there is no present bias, and quasi-hyperbolic discounting is identical to exponential discounting. When $\beta < 1$, the smaller the β, the smaller weight the agent attaches to the time that is 'the future', and hence the greater the present bias is. When $\beta = 0$, the agent only cares about the payment today. In addition to the fact that the agent is present-biased, the agent is also assumed to be aware of his present bias. That is, the agent knows that when the *future* becomes the *present*, his preference changes. This type of the agent is referred to as the *sophisticated* agent in the literature.

In this section, I show that the optimal strategy for the present-biased agent with the discount parameter $\delta \leq \bar{\delta}$ and the present-biased parameter $\beta \in (0,1)$ is still the Linear Search Strategy. This is because the present-biased preference gives the agent a stronger incentive to choose the Linear Search Policy in each state. Since it is already optimal for his time-consistent counterpart to choose the Linear Search Policy in each state, it is hence also optimal for the present-biased agent to choose the Linear Search Policy in each state when $\delta \leq \bar{\delta}$.

For the present-biased agent with $\delta > \bar{\delta}$, the optimal strategy depends on the agent's present-biased parameter. When the agent is not very present-biased, the Binary Search Strategy is still the optimal strategy. Suppose the agent's future selves use the Binary Search Policy in all future states. First notice that when $\delta > \bar{\delta}$, any $(m,n) \in \mathcal{F}^\dagger$ policy gives a weakly smaller payoff than that of the Binary Search Policy in state N. This is because $(m,n) \in \mathcal{F}^\dagger$ policy and Binary Search Policy give the agent zero immediate reward. Since the present-biased agent is assumed to be consistent when evaluating the future payoffs, she behaves essentially the same as the time-consistent agent when there is no immediate payoff getting involved. As a result, I only consider whether it is optimal for the agent to use the Linear Search Policy in the current state. If the agent uses the Linear Search Policy in the current state, the benefit is the immediate expected reward associated with learning. That is $\frac{1}{N}$. The (future) cost is the difference between using the Binary Search today and tomorrow $V^B(N) - \delta\frac{N-1}{N}V^B(N-1)$. If the agent is not very present-biased, that is, the present-biased parameter is greater than the ratio of the benefit to the cost, then, using the Linear Search Policy in current state N is not optimal. Using the Binary Search Policy in state N hence is optimal, given that her future selves use the Binary Search Policy. Using the idea of backward induction, one can check whether using the Binary Search Policy is optimal in state N given that the agent's future selves use the Binary Search Policy. If this is true for all the states, then, the Binary Search Strategy is the optimal strategy. This requires the agent's present-biased parameter to be greater than $\bar{\beta}^N := \frac{\frac{1}{N}}{V^B(N)-\delta\frac{N-1}{N}V^B(N-1)}$ for $N = 4, 5, \ldots, \bar{N}$ where $\bar{N}$ is the cardinality of the parameter set. The benefit associated with the Linear Search Policy is the highest in state $N = 4$, and the cost associated with the Binary Search Policy is the lowest in state $N = 4$. As a result, if the present-biased parameter β is weakly greater than $\check{\beta} := \bar{\beta}^4$, then, the Binary Search Strategy is the optimal strategy.

When $\delta > \bar{\delta}$, for any $\beta < \check{\beta}$, there exists a state $\underline{N}$ such that the Linear Search Policy is optimal in all the states $H < \underline{N}$, and the Binary Search Policy is optimal in state $\underline{N}$. That is, there exists a switch of actions from Binary Search (when the state is big) to Linear Search (when the state becomes small). Suppose the agent's future selves use the Linear Search Policy in all future states. In order to check which policy is optimal in state N, let

$$P(N) \equiv \max_{\{m,n\}\in\mathcal{F}^\dagger} \delta\left\{\frac{m}{N}V^L(m) + \frac{n}{N}V^L(n)\right\}$$

be the highest value associated with the policy $(m, n) \in \mathcal{F}^\dagger$ in state N given that the agent's future selves use the Linear Search Policy in all the states smaller than N. If the agent uses the Linear Search Policy in the current state, the benefit is the immediate expected reward associated with learning. That is $\frac{1}{N}$. The cost of using the Linear Search Policy in state N is the forgone payoff from using the $(m, n) \in \mathcal{F}^\dagger$ policy, which is $P(N) - \delta\frac{N-1}{N}V^L(N-1)$. Using the idea of backward induction, one can check which Policy is optimal in state N given that the agent's future selves use the Linear Search Policy. If the agent's present-biased parameter is smaller than $\tilde{\beta}^N := \frac{\frac{1}{N}}{P(N)-\delta\frac{N-1}{N}V^L(N-1)}$, the ratio of the benefit to the cost of using the Linear Search Policy in state N, then, it is optimal for the agent to use the Linear Search Policy in state N. Otherwise, it is optimal for her to use the $(m, n) \in \mathcal{F}^\dagger$ policy that achieves $P(N)$ in state N. From the discussion in the benchmark case,we know that when $\delta > \bar{\delta}$, the $(m, n) \in \mathcal{F}^\dagger$ policy that achieves $P(N)$ in state N is the Binary Search Policy because of the concavity of the function $W^L(N) = NV^L(N)$. Therefore, when the agent's present-biased parameter is smaller than $\tilde{\beta}^N$, it is optimal for the agent to use the Binary Search Policy in state N.

When it is optimal for the agent to use the Binary Search Policy in a state $\underline{N}$ given the belief that her future selves use the Linear Search Policy, we still need to check whether it is indeed optimal for her future selves to use the Linear Search Policy. In the appendix, I show that $\tilde{\beta}^N$ decreases in N. This is because the benefit of using the Linear Search Policy in state N decreases in N, and the cost increases in N. As a result, in order to guarantee that it is optimal for the agent to use the Linear Search Policy in any state smaller than $\underline{N}$, the agent's present-biased parameter should be greater than $\tilde{\beta}^{\underline{N}-1}$.

The following proposition concludes the result discussed above.

Proposition 4. *When the present-biased agent has the discount parameter $\delta \leq \bar{\delta}$ and the present-biased parameter $\beta \in (0, 1)$, the optimal strategy is the Linear Search Strategy.*

When the present-biased agent has the discount parameter $\delta > \bar{\delta}$ and the present-biased parameter $\beta \geq \check{\beta}$, the optimal strategy is the Binary Search Strategy.

When the present-biased agent has the discount parameter $\delta > \bar{\delta}$ and the present-biased parameter $\beta \in [\tilde{\beta}^{\underline{N}}, \tilde{\beta}^{\underline{N}-1})$, it is optimal for the agent to use the Binary Search Policy in state $\underline{N}$ and to use the Linear Search Policy in state $H < \underline{N}$.

Consider an agent with discount parameter $\delta > \bar{\delta}$ and the present-biased parameter $\beta \in [\tilde{\beta}^{\underline{N}}, \tilde{\beta}^{\underline{N}-1})$, the last point of Proposition 4 indicates that in states $N < \underline{N}$, the agent's optimal policy is the Linear Search Policy, and the agent's optimal policy in state $\underline{N}$ is the Binary Search Policy. But in states $N > \underline{N}$, we do not know what the optimal policy of the agent is. Proposition 4 does not characterise the complete optimal strategy when $\delta > \bar{\delta}$ and $\beta < \tilde{\beta}^4$ due to the complexity of the calculation. Instead, it only specifies that there exists at least one switch between the Linear Search Policy and the Binary Search Policy. If there exists more than one switch between actions, the last point of Proposition 4 characterises the smallest state in which the agent switches actions. This serves the purpose to show that when the agent has time-inconsistent preferences, the optimal strategy does not have the consistent property as in the cases with time-consistent preferences.

To better explain the result, I introduce the *present-biased agent's time-consistent counterpart.* If a present-biased agent has a discount parameter $\hat{\delta}$ and a present-biased parameter $\hat{\beta}$, then his time-consistent counterpart also has a discount parameter $\hat{\delta}$, but a present-biased parameter $\beta = 1$. The result above indicates that in comparison with the present-biased agent's time-consistent counterpart, it is optimal for the present-biased agent to use the Linear Search Policy in small states, while it is *always* optimal for his time-consistent counterpart to use the Binary Search Policy. This is because the value associated with a policy in state N consists of two parts: the expected payment today and the perceived discounted continuation value. Given the future policies remaining the same, consider the cost and the benefit of the Linear Search Policy in the current state N. The cost of the Linear Search Policy comes from the discounted continuation value. The greater the current state N, the more costly to use the Linear Search Policy. The benefit of the Linear Search Policy is from the expected payment today. The smaller the state N, the greater the benefit associated with the Linear Search Policy. Notice that when the agent is present-biased, the cost of the Linear Search is perceived to be smaller because of the present-biased parameter, but she evaluates the expected payment today the same as his time-consistent counterpart. As a result, there are two driving forces that incentivise the agent to use the Linear Search Policy: one is that the state is sufficiently small and thus the benefit of Linear Search becomes higher, and the other one is that the agent is sufficiently present-biased and thus perceives the cost of Linear Search to be smaller. The main tradeoff is that when the agent is very present-biased, that is, she has a smaller present-biased parameter β, she perceives the Linear Search Policy to be attractive in bigger states, while for the agent who is less present-biased, she only perceives the Linear Search Policy to be attractive in smaller states.

7 Winner takes all

In the previous section, I have discussed a single agent's decision problem. There is only one agent learning the unknown parameter. In this section, I consider a game where the players compete to learn the unknown parameter. At each time t, the player $i \in \{1, 2, \ldots, I\}$ is active with probability p_i where $\sum_i^I p_i = 1$ and $p_i \in (0, 1)$ for all i. There is only one active player at each time t. The active player i can use the Linear Search Policy, the Binary Search Policy, or any $(m, n) \in \mathcal{F}^\dagger$ policy in state N. At time t, if the active player learns the unknown parameter, then, she gets the reward associated with learning and the game ends. All other players get no reward. If the active player does not learn the unknown parameter, the game enters next period. Assume the lifetime utility of the players is as in eq. (2) and all the players have the same discount parameter δ.

If all the players use the Linear Search Strategy,then, the outcome of the game, that is, the distribution of the learning time, is the same as the case with one player learning alone using the Linear Search Strategy. The players' payoffs are different because now they only have the chance to get the reward associated with learning when they are active. One could imagine that now the players have less incentives to use the Binary Search Strategy because it gives the player zero probability of getting the reward in state greater than two. In addition, using the Binary Search Strategy increases the other players' probabilities of getting the reward by shrinking the parameter set. It is then interesting to check when it is an equilibrium such that all the players use the Linear Search Strategy I show that this equilibrium exists when the discount parameter δ is not too close to one.

Proposition 5. *There exists an upperbound $\mathring{\delta} = \min \frac{1}{1+p_i}$ such that if $\delta \leq \mathring{\delta}$, there exists an equilibrium where all the players use the Linear Search Strategy.*

Suppose all the players use the Linear Search Strategy and then I check whether there exists any deviation in state N that gives the player a higher payoff. First notice that, when $\delta \leq \frac{1}{2}$, the player does not want to deviate to other policies in any state. This is because when players are competing to get the reward associated with learning, the players have stronger incentives to use the Linear Search Strategy. If it is optimal for the player to use the Linear Search Strategy when she is learning alone, it is still optimal for her to use the Linear Search Strategy when competing with other players. When $\delta > \frac{1}{2}$, the player faces a tradeoff where the Linear Search Policy gives the player the probability of getting the immediate payoff, while using other policies reduces the future state and increases her payoff when she is active again in the future. Which effect dominates depends on the player's discount parameter and the probability that she is active in each state. If the player is active with a high probability, say, p_i is close to one, then, the Linear Search Strategy best responses to other players using the Linear Search Strategy only when δ is close to $\frac{1}{2}$. If the player is active with a probability close to zero, then, the Linear Search Strategy best responses to other players using the Linear Search Strategy even when δ is close to one. If the best response of the most active player (i.e. the player with the highest probability of being active) to the Linear Search Strategy is the Linear Search Strategy, then, it is an equilibrium where all the players use the Linear Search Strategy.

When p_i is smaller than one for all i, the upperbound $\mathring{\delta}$ is greater than a half. This shows that the sufficiently patient players (whose optimal strategy is the Binary Search Strategy when learning alone) can switch to the Linear Search Strategy when competing with others. This leads to an inefficient outcome because it takes longer to learn the parameter in expectation when the players use the Linear Search Strategy. This inefficiency prevails for a larger range of the discount parameter when the players are active with equal probabilities. In addition, if the players are active with equal probabilities, increasing the number of players intensifies the inefficiency. That is, all the players using the Linear Search Strategy is an equilibrium for a larger range of the discount parameter. Suppose there are two players who are active in each state with probability 0.9 and 0.1 respectively. Then, the upperbound $\mathring{\delta}$ is 0.53. Thus, both players using the Linear Search Strategy is an equilibrium when the discount parameter δ is weakly smaller than 0.53. If both players are active in each state with probability a half, then, the upperbound $\mathring{\delta}$ is 0.67. In this case, both players using the Linear Search Strategy is an equilibrium when the discount parameter δ is weakly smaller than 0.67. When the players are active with equal probability, both players using the Linear Search Strategy is an equilibrium for a greater range of the discount parameter. When there are I players and all the players are active with equal probability in each state, we have $p_i = \frac{1}{I}$. Since the upperbound $\mathring{\delta} = \frac{I}{I+1}$ increases in I, all the players using the Linear Search Strategy is an equilibrium for a larger discount parameter set when I increases. Increasing the number of players hence intensifies the inefficiency.

A Appendix

A.1 Search with a fixed cost

In this section, the agent does not discount future payoffs with a discount parameter. Instead, she pays a fixed cost $c > 0$ for each search if she does not learn the true parameter. As mentioned in Section 3, the agent's preference is time risk neutral. The agent's optimal strategy is the Binary Search Strategy as long as the fixed cost c is bounded. Otherwise, the agent does not search. The main intuition is that since the agent has to pay a fixed cost c for each search, the agent's only incentive is to learn with the least number of searches. The Binary Search Strategy satisfies this requirement. However, since the agent only gets a reward of one associated with learning, she is thus not willing to search when the fixed cost c is too high. Let $\bar{N}$ denote the initial state, which is the cardinality of the parameter set Θ. Let $\bar{c}(\bar{N}) \equiv \frac{\bar{N}}{(2\bar{N}-2^{\lfloor \log_2 \bar{N} \rfloor+1})(\lceil \log_2 \bar{N} \rceil-1)+(2^{\lfloor \log_2 \bar{N} \rfloor+1}-\bar{N})(\lfloor \log_2 \bar{N} \rfloor-1)}$.

Proposition 6. *Given any initial state $\bar{N}$, if the fixed cost $c \leq \bar{c}(\bar{N})$, the agent searches and the optimal strategy is the Binary Search Strategy. Otherwise, the agent does not search. The Linear Search Strategy is always sub-optimal.*

The idea of the proof is the same as in the benchmark case: compare the lifetime utility of the Binary (Linear, resp) Search Strategy and the lifetime utility of the Binary (Linear, resp) Search Deviating Strategies. There is always a Linear Search Deviating Strategy that gives a higher lifetime utility than the Linear Search Strategy, and no Binary Search Deviating Strategy gives a higher lifetime utility than the Binary Search Strategy.

A.2 Relax the uniform prior assumption

Note that the agent's prior belief is $f_0 = (f_0(\theta))_{\theta\in\Theta}$ with $f_0(\theta) \in (0, 1)$ for $\forall\theta \in \Theta$. In the previous discussion, f_0 is assumed to be uniform. That is, $f_0(\theta) = \frac{1}{\bar{N}}$ for $\forall\theta \in \Theta$, where $\bar{N}$ is the cardinality of the parameter set Θ. In this section, I assume that the agent believes that θ_1, the first element in the parameter set, is most likely to be the true parameter, whereas all other parameters in the parameter set are equally likely. That is $f(\theta_1) = f_1$, and $f(\theta) = \frac{1-f_1}{N-1}$ for $\theta \in \Theta \setminus \{\theta_1\}$. I refer to this distribution as a *pseudo-uniform distribution with a peak of* f_1. To keep the model simple, I consider the preference in the benchmark case: expected utility with exponential discounting. In addition, I make the following assumption.

Assumption 1. *The probability attached to the element θ_1 is not smaller than a half, which is $f_1 \geq \frac{1}{2}$.*

This assumption guarantees that any revised belief of the agent is either a uniform distribution or a distribution with a *peak*. If this assumption is violated, the revised belief may not describe the fact that the agent believes the first element is *most* likely to be the true parameter, but rather the first element is *least* likely to be the true parameter. This assumption allows me to retain the simplification of the agent's choice in each state. In the previous discussion, the feasible set of the choice in state N is denoted by $\mathcal{F} = \{(m, n)|m+n = N, m, n \in \mathbb{Z}^+ \cap [1, N]\}$. In the benchmark case, due to the symmetry of the uniform distribution, assuming $m \leq n$ is without loss of generality. In this section, with the assumption that $f_1 \geq \frac{1}{2}$, choosing $m > \frac{N}{2}$ is always sub-optimal. For example, if N is even, choosing $m = \frac{N}{2} + 1$ is dominated by choosing $m' = \frac{N}{2} - 1$. The simplification that $m \leq n$ is still without loss of generality.

The main question of interest in this section is to find the conditions under which *the Focal Point Search Strategy* (defined below) is optimal.

Definition 4. *The Focal Point Search Strategy* is a strategy such that (1) in state N, if the belief of the agent has a peak, then the agent chooses $(m, n) = (1, N - 1)$, (2) in state N, if the belief of the agent is uniform, then the agent chooses the Linear Search Policy if $\delta \leq 0.5$ and the Binary Search Policy if $\delta > 0.5$.

When $\delta \leq 0.5$, the Focal Point Search Strategy coincides with the Linear Search Strategy. When $\delta > 0.5$, the Focal Point Search Strategy can be considered as the agent uses the Linear Search Policy when the belief is not uniform, and then switch to the Binary Search Strategy when the belief becomes uniform. The agent with $\delta > 0.5$ is only willing to use the Linear Search Policy under one situation: when she is quite certain that the first element in the parameter set is the true value of the parameter. Hence, asking the question that under what conditions the Focal Point Search Strategy is optimal, is equivalent to asking how certain the agent has to be so that she is willing to test the first element in the parameter set before doing anything else.

Proposition 7. *When $\delta \leq 0.5$, the Focal Point Search Strategy is optimal.*

The intuition is that when the agent uses the Linear Search Strategy, the agent's utility is a convex combination of the payoff of learning the unknown parameter and the discounted continuation value. The continuation value is the same under uniform prior and under the prior with a peak, and it is smaller than the payoff of learning the unknown parameter. When the agent's prior belief is the distribution with a peak, the agent puts higher weight to the payoff of learning the unknown parameter today. Therefore, if it is optimal for the agent to use the Linear Search Strategy when the prior is uniform, it is also optimal for the agent to use the Focal Point Search Strategy when the prior is the distribution with a peak.

Proposition 8. *If $\delta \in (0.5, f_1]$, then the Focal Point Search Strategy is optimal.*

This proposition says that given the value of f_1, when the agent is not too patient, that is, when $\delta \leq f_1$, the Focal Point Search Strategy is optimal. The other way to interpret this proposition is that given the value of the agent's discount parameter $\delta > \frac{1}{2}$, if the agent is sufficiently certain that the first element is the true value of the parameter, that is, $f_1 \geq \delta$, then, he/she is willing to use the Linear Search Policy to check the first element in the parameter set, and then uses the Binary Search Strategy if the first element is not the true value of the parameter.

This proposition provides a sufficient condition under which the Focal Point Search Strategy is optimal. It does not characterise the optimal strategy when $\delta > f_1$. Even with this seemingly simple prior assumption, it is still hard to get the characterisation of the optimal strategy for any given value of δ and f_1. However, the optimal strategy under some extreme cases may be helpful to understand the agent's behaviour. When the agent has $\delta = 1$, she is indifferent between the Linear Search Strategy, the Binary Search Strategy and the Focal Point Search Strategy because all the future payoffs are perceived to be the same. Whenever and however the agent learns the unknown parameter, he/she gets a payoff of one. This shows that increasing δ does not monotonically increase the agent's incentive to use the Binary Search Strategy, otherwise, when δ takes the highest value, the optimal strategy must be unique: the Binary Search Strategy. I may make the *conjecture* that when δ is close to one, it might be optimal for the agent to use the Focal Point Search Strategy. The intuition is that when the agent is very patient, the agent does not mind using one period of Linear Search Policy to test the most likely element. But, this is just a conjecture, there is yet no proof to support this conjecture.

B Proofs

B.1 The proof of Lemma 1

The value associated with the Linear Search Strategy is

$$\begin{aligned}
V^L(N) &= \frac{1}{N} + \delta\frac{N-1}{N}V^L(N-1) \\
&\cdots \\
&= \frac{1}{N} + \delta\frac{1}{N} + \delta^2\frac{1}{N} + \cdots + \delta^{N-2}\frac{2}{N}V^L(2) \\
&= \frac{1}{N}\Big(\sum_{i=0}^{N-2}\delta^i + \delta^{N-2}\Big) = \frac{1}{N}\Big(\frac{1-\delta^{N-1}}{1-\delta} + \delta^{N-2}\Big)
\end{aligned}$$

To derive the value associated with the Binary Search Strategy, I use mathematical induction.

To simplify the notation, I rewrite the value in the following way

$$NV^B(N) = \pi^N \delta^{\tau_1^N} + (N - \pi^N)\delta^{\tau_2^N}. \tag{9}$$

I also establish the properties of π^N and τ^N.

Remark 1. *The following equations hold.*

$$\begin{aligned}
\pi^{2N} &= 2\pi^N \\
\tau_i^{2N} &= \tau_i^N + 1 \textit{ for } i = 1, 2
\end{aligned}$$

I first show that if eq. (9) is the value function in state N associated with the Binary Search Strategy, then we have $2NV^B(2N)$, the value function in state $2N$ associated with the Binary Search Strategy, following the same functional form as eq. (9). The calculation is as follows.

$$\begin{aligned}
2NV^B(2N) &= \delta\Big\{NV^B(N) + NV^B(N)\Big\} \\
&= 2\pi^N\delta^{\tau_1^N+1} + (2N - 2\pi^N)\delta^{\tau_2^N+1} \\
&= \pi^{2N}\delta^{\tau_1^{2N}} + (2N - \pi^{2N})\delta^{\tau_2^{2N}}
\end{aligned}$$

Before showing the next part of the proof, I establish some properties of π^N and τ^N. Let $K \in \mathbb{Z}^+$.

Remark 2. *If $N = 2^K$, the following equations hold.*

$$\begin{aligned}
\pi^{2N+1} &= \pi^N + \pi^{N+1} = \pi^{N+1} \\
\tau_1^{2N+1} &= \tau_1^{N+1} + 1 = \tau_1^N + 2 \\
\tau_2^{2N+1} &= \tau_2^{N+1} + 1 = \tau_2^N + 1
\end{aligned}$$

If $N \in [2^K + 1, 2^{K+1} - 2]$, the following equations hold.

$$\begin{aligned}
\pi^{2N+1} &= \pi^N + \pi^{N+1} \\
\tau_i^{2N+1} &= \tau_i^N + 1 = \tau_i^{N+1} + 1 \textit{ for } i = 1, 2
\end{aligned}$$

If $N = 2^{K+1} - 1$, the following equations hold.

$$\begin{aligned}
&N - \pi^N = (2N + 1) - \pi^{2N+1} \\
&\pi^N = N - 1;\ \pi^{2N+1} = 2N;\ \pi^{N+1} = 0 \\
&\tau_1^N = \tau_1^{N+1} = \tau_2^{N+1} \\
&\tau_1^{2N+1} = \tau_1^N + 1 = \tau_1^{N+1} + 1 \\
&\tau_2^{2N+1} = \tau_2^N + 1 = \tau_2^{N+1}
\end{aligned}$$

Next, I show that If the value function in state N associated with the Binary Search Strategy follows the functional form eq. (9), and the value function in state $N + 1$ associated with the

Binary Search Strategy follows the functional form eq. (9), then, the value function in state $2N+1$ associated with the Binary Search Strategy follows the same function form as eq. (9). The calculation is as follows.

$$
\begin{aligned}
(2N+1)V^B(2N+1) &= \delta\Big\{NV^B(N) + (N+1)V^B(N+1)\Big\} \\
&= \pi^N\delta^{\tau_1^N+1} + (N-\pi^N)\delta^{\tau_2^N+1} + \pi^{N+1}\delta^{\tau_1^{N+1}+1} + (N+1-\pi^{N+1})\delta^{\tau_2^{N+1}+1}
\end{aligned}
$$

If $N = 2^K$, then

$$
\begin{aligned}
(2N+1)V^B(2N+1) &= N\delta^{\tau_2^N+1} + \pi^{N+1}\delta^{\tau_1^{N+1}+1} + (N+1-\pi^{N+1})\delta^{\tau_2^{N+1}+1} \\
&= N\delta^{\tau_2^{2N+1}} + \pi^{2N+1}\delta^{\tau_1^{2N+1}} + (N+1-\pi^{2N+1})\delta^{\tau_2^{2N+1}} \\
&= \pi^{2N+1}\delta^{\tau_1^{2N+1}} + (2N+1-\pi^{2N+1})\delta^{\tau_2^{2N+1}}
\end{aligned}
$$

follows the same functional form as eq. (9). If $N \in [2^K+1, 2^{K+1}-2]$, then

$$
\begin{aligned}
(2N+1)V^B(2N+1) &= \pi^N\delta^{\tau_1^{2N+1}} + (N-\pi^N)\delta^{\tau_2^{2N+1}} + \pi^{N+1}\delta^{\tau_1^{2N+1}} + (N+1-\pi^{N+1})\delta^{\tau_2^{2N+1}} \\
&= \pi^{2N+1}\delta^{\tau_1^{2N+1}} + (2N+1-\pi^{2N+1})\delta^{\tau_2^{2N+1}}
\end{aligned}
$$

follows the same functional form as eq. (9). If $N = 2^{K+1}-1$, then

$$
\begin{aligned}
(2N+1)V^B(2N+1) &= \pi^N\delta^{\tau_1^N+1} + (N-\pi^N)\delta^{\tau_2^N+1} + (N+1)\delta^{\tau_2^{N+1}+1} \\
&= (N-1)\delta^{\tau_1^{2N+1}} + (2N+1-\pi^{2N+1})\delta^{\tau_2^{2N+1}} + (N+1)\delta^{\tau_1^{2N+1}} \\
&= \pi^{2N+1}\delta^{\tau_1^{2N+1}} + (2N+1-\pi^{2N+1})\delta^{\tau_2^{2N+1}}
\end{aligned}
$$

follows the same functional form as eq. (9).

Lastly, I show that when $N \in \{3,4,5\}$, the value function in state N associated with Binary Search follows the same functional form as eq. (9). If $N=3$, $3V^B(3) = 1+2\delta$ follows eq. (9). If $N=4$, $4V^B(4) = 4\delta$ follows eq. (9). If $N=5$, $5V^B(5) = 3\delta + 2\delta^2$ follows eq. (9).

B.2 The proof of Lemma 2

To show the properties of $W^L(N)$, It is without loss of generality to treat N as a continuous variable and then compute the first and second order derivatives. The first order derivative of $W^L(N)$ is $(\log\delta)\frac{1-2\delta}{1-\delta}\delta^{N-2}$, and the second order derivative of $W^L(N)$ is $(\log^2\delta)\frac{1-2\delta}{1-\delta}\delta^{N-2}$. When $\delta < \frac{1}{2}$, the first order derivative of $W^L(N)$ is negative and the second order derivative is positive. When $\delta > \frac{1}{2}$, the first order derivative of $W^L(N)$ is positive and the second order derivative is negative. When $\delta = \frac{1}{2}$, $W^L(N)$ is independent of N.

The first and second order difference of $W^B(N)$ is summarised in the following lemma. Let $\triangle W^B(N) := W^B(N) - W^B(N-1)$ and $\triangle^2 W^B(N) := \triangle W^B(N) - \triangle W^B(N-1)$.

Lemma 6. *If $N = 2^Y$, then, $\triangle W^B(N) = \delta^{Y-2}(2\delta-1)$ for all $Y \in \mathbb{Z}^+$.*

If $2^Y < N < 2^{Y+1}$, then, $\triangle W^B(N) = \delta^{Y-1}(2\delta-1)$ for all $Y \in \mathbb{Z}^+$.

If $N = 2^Y$ and $2^{Y-1} < N-1 < 2^Y$, then, $\triangle^2 W^B(N) = 0$.

If $2^Y < N < 2^{Y+1}$ *and* $2^Y < N-1 < 2^{Y+1}$*, then,* $\triangle^2 W^B(N) = 0$.
If $2^Y < N < 2^{Y+1}$ *and* $N-1 = 2^Y$*, then,* $\triangle^2 W^B(N) = (2\delta-1)(\delta-1)\delta^{Y-2}$.

Lemma 6 is derived directly from calculating the first difference of the function $W^B(\cdot)$. When $\delta > 0.5$ ($\delta < 0.5$), The first-order difference of $W^B(\cdot)$ is positive (negative), and the second-order difference of $W^B(\cdot)$ is non-positive (non-negative). When $\delta = 0.5$, $W^B(N)$ is independent of N.

B.3 The proof of Proposition 1

To show that the Linear Search Strategy is optimal, I show that there is no Linear Search Deviating Strategy give the agent a higher payoff than that of the Linear Search Strategy. In state N, the value associated with the Linear Search Deviating Strategy is

$$\begin{aligned}
V^D(N) &= \delta\Big\{\frac{m}{N}V^L(m) + \frac{n}{N}V^L(n)\Big\} \\
&= \delta\Big[\frac{m}{N}\frac{1}{m}\Big(\frac{1-\delta^{m-1}}{1-\delta} + \delta^{m-2}\Big) + \frac{n}{N}\frac{1}{n}\Big(\frac{1-\delta^{n-1}}{1-\delta} + \delta^{n-2}\Big)\Big] \\
&= V^L(N)\frac{2\delta + (1-2\delta)(\delta^{m-1}+\delta^{n-1})}{2\delta + (1-2\delta)(1+\delta^{N-2})}
\end{aligned}$$

Let $M = \frac{2\delta+(1-2\delta)(\delta^{m-1}+\delta^{n-1})}{2\delta+(1-2\delta)(1+\delta^{N-2})}$. I show that $M \leq 1$ when $\delta \leq 0.5$ and $M > 1$ when $\delta > 0.5$. Notice that $2\delta + (1-2\delta)(\delta^{m-1}+\delta^{n-1}) > 0$ and $2\delta + (1-2\delta)(1+\delta^{N-2}) > 0$. I calculate the differences of the denominator and numerator of M.

$$\begin{aligned}
&\Big[2\delta + (1-2\delta)(\delta^{m-1}+\delta^{n-1})\Big] - \Big[2\delta + (1-2\delta)(1+\delta^{N-2})\Big] \\
&= (1-2\delta)\Big[(\delta^{m-1}+\delta^{n-1}) - (1+\delta^{m+n-2})\Big]
\end{aligned}$$

Notice that $(\delta^{m-1}+\delta^{n-1})-(1+\delta^{m+n-2}) = -(1-\delta^{n-1})(1-\delta^{m-1}) < 0$ Therefore, $2\delta+(1-2\delta)(\delta^{m-1}+\delta^{n-1}) > 2\delta+(1-2\delta)(1+\delta^{N-2})$ when $\delta > 0.5$ and $2\delta+(1-2\delta)(\delta^{m-1}+\delta^{n-1}) \leq 2\delta+(1-2\delta)(1+\delta^{N-2})$ when $\delta \leq 0.5$. Therefore, $M \leq 1$ when $\delta \leq 0.5$ and $M > 1$ when $\delta > 0.5$. Thus, $V^L(N) \geq V^D(N)$ if $\delta \leq 0.5$. As a result, a *Linear Search Deviating Strategy* that gives the agent a higher payoff than the Linear Search Strategy does not exist.

To show that the Binary Search Strategy is optimal, I first show that the Binary Search Deviating Strategy such that the $(m,n) \in \mathcal{F}^\dagger$ policy in state N is used gives the agent a lower payoff than the Binary Search Strategy. Since I assume that $m \leq n$, the Binary Search Policy in state N by definition maximises m and minimises n. To show that deviating to other choices $(m,n) \in \mathcal{F}^\dagger$ is not profitable, I show that

$$\delta\Big\{mV^B(m) + nV^B(n)\Big\} \geq \delta\Big\{(m-1)V^B(m-1) + (n+1)V^B(n+1)\Big\}$$

if $\delta \geq 0.5$.

Rearrange the inequality, it is equivalent to show that

$$\triangle W^B(m) \geq \triangle W^B(n+1). \tag{10}$$

According to Lemma 6, I consider four cases based on the values of m and n.

Case 1: $2^Y < m < 2^{Y+1}$ and $2^K - 1 < n < 2^{K+1} - 1$. Since $m \leq n$, $\log_2 m \leq \log_2 n$. Then, $\lfloor \log_2 m \rfloor \leq \lfloor \log_2 n \rfloor$, which is $Y \leq K$. Therefore,

$$\delta^{Y-1}(2\delta - 1) \geq \delta^{K-1}(2\delta - 1).$$

The inequality (10) holds if $\delta \geq 0.5$.

Case 2: $m = 2^Y$ and $n = 2^K - 1$. Since $m \leq n$, we have $Y \leq K - 1$. Thus $Y - 2 < K - 2$ Therefore,

$$\delta^{Y-2}(2\delta - 1) > \delta^{K-2}(2\delta - 1).$$

The inequality (10) holds if $\delta \geq 0.5$.

Case 3: $2^Y < m < 2^{Y+1}$ and $n = 2^K - 1$. Since $m \leq n$, we have $Y \leq K - 1$. Therefore

$$\delta^{Y-1}(2\delta - 1) \geq \delta^{K-2}(2\delta - 1).$$

The inequality (10) holds if $\delta \geq 0.5$.

Case 4: $m = 2^Y$ and $2^K - 1 < n < 2^{K+1} - 1$. Since $m \leq n$, we have $Y \leq K$. Therefore,

$$\delta^{Y-2}(2\delta - 1) > \delta^{K-1}(2\delta - 1).$$

The inequality (10) holds if $\delta \geq 0.5$.

Next I show that deviating to Linear Search in state N is not profitable. Before showing the result, I introduce the following lemma.

Lemma 7. *If $\delta \in [0.5, 1)$, $W^B(N) - \delta W^B(N-1)$ weakly increases in N.*

This lemma can be shown given Lemma 6

To show that deviating to Linear Search in state N is not profitable, I show that

$$\frac{1}{N} + \frac{N-1}{N}\delta V^B(N-1) \leq V^B(N),$$

which is equivalent to

$$1 \leq NV^B(N) - \delta(N-1)V^B(N-1).$$

Given Lemma 7, the minimum of $W(N)$ is 1. Since $W(N)$ is increasing, $W(N) \geq 1$. Therefore, deviating to Linear Search in state N is not profitable if $\delta \geq 0.5$.

To summarise, there is no Binary Search Deviating strategy that gives the agent a higher payoff than the Binary Search Strategy if $\delta \geq 0.5$.

B.4 The proof of Lemma 4

In this proof, I find $\phi(\cdot)$ and $u(\cdot)$ such that $\phi\big(\delta^\tau u(x)\big) = \rho^\tau x$. Let $z = \delta^\tau$ and $u(x) = y$. Then,

$$\phi\big(\delta^\tau u(x)\big) = \phi(zy) = z^{\frac{\log \rho}{\log \delta}} u^{-1}(y).$$

Let $t = zy$. Then,

$$\phi(t) = u^{-1}(y)\left(\frac{1}{y}\right)^{\frac{\log\rho}{\log\delta}} t^{\frac{\log\rho}{\log\delta}} := at^{\frac{\log\rho}{\log\delta}}$$

where $a := u^{-1}(y)\left(\frac{1}{y}\right)^{\frac{\log\rho}{\log\delta}}$. Given the functional form of $\phi(\cdot)$, next, I find the functional form of $u(\cdot)$ such that

$$a\left[\delta^{\tau}u\left(x\right)\right]^{\frac{\log\rho}{\log\delta}} = \rho^{\tau}x.$$

Then, we have $u(x) = \left(\frac{x}{a}\right)^{\frac{\log\delta}{\log\rho}}$. Let $c := \frac{\log\rho}{\log\delta}$. Then, $\phi(x) = ax^c$ and $u(x) = \left(\frac{x}{a}\right)^{\frac{1}{c}}$.

B.5 The proof of Proposition 3

Let $T(\mathcal{W})(N) = \max_{(m,n)\in\mathcal{F}}\zeta\left(\frac{m}{N}\mathcal{W}(m)+\frac{n}{N}\mathcal{W}(n)\right)$ and $\mathcal{B}(\mathbb{Z}^+)$ be a space of the bounded functions $\mathcal{W}$: $\mathbb{Z}^+ \to \mathbb{R}$. The operator $T(\mathcal{W})$ is a contraction mapping maps from $\mathcal{B}(\mathbb{Z}^+)$ to $\mathcal{B}(\mathbb{Z}^+)$. The fixed point can be derived based on the discussion in the benchmark case. If $\zeta \in (0, \bar{\zeta})$, the fixed point of the mapping is $\mathcal{W}(N) = V^L(N)$. The corresponding strategy is the Linear Search Strategy. If $\zeta \in (\bar{\zeta}, 1)$, the fixed point of the mapping is $\mathcal{W}(N) = V^B(N)$. The corresponding strategy is the Binary Search Strategy. Since $\alpha \in (0, 1)$, the value function $\mathcal{E}(N)$ is maximised when $\mathcal{W}(N)$ is maximised. Therefore, if $\zeta \in (0, \bar{\zeta})$, the optimal strategy to achieve the maximum $\mathcal{E}(N)$ is the Linear Search Strategy. If $\zeta \in (\bar{\zeta}, 1)$, the optimal strategy to achieve the maximum $\mathcal{E}(N)$ is the Binary Search Strategy. Since $\zeta = \delta^{\frac{\alpha}{\rho}}$, given any (α, ρ) pair, there exists a unique threshold $\tilde{\delta} = (\bar{\zeta})^{\frac{\rho}{\alpha}}$ such that if $\delta > \tilde{\delta}$, the Binary Search Strategy is the optimal strategy. If $\delta < \tilde{\delta}$, the Linear Search Strategy is the optimal strategy.

B.6 The proof of Proposition 4

Before showing the proofs of the propositions, I first introduce some definitions that will be used in the proofs.

Given a strategy, the agent's state-N self's value function associated with that strategy is related to the value functions of the time-consistent agent. For example, the agent's state-N self's value associated with the Linear Search Strategy $\mathcal{U}^L(N)$ is

$$\mathcal{U}^L(N) = \frac{1}{N} + \beta\delta\frac{N-1}{N}V^L(N-1),$$

where $V^L(\cdot)$ is the value associated with the Linear Search Strategy in the benchmark case (see Lemma 1). This is because the present-biased agent perceives all the future payments to be less important than the payment at present. This idea is formally characterised in the following lemma. Let $\mathcal{S}$ be a strategy and let $\mathcal{U}^{\mathcal{S}}(N)$ be the agent's state-N self's value function associated with this strategy. Let $V^{\mathcal{S}}(\cdot)$ be the time-consistent agent's value function associated with the strategy $\mathcal{S}$, and let $(m^{\mathcal{S}}, n^{\mathcal{S}})$ denote the strategy $\mathcal{S}$ induced policy in state N.

Lemma 8. *The present-biased agent's state-N self's value function associated with the strategy $\mathcal{S}$*

is

$$\mathcal{U}^{\mathcal{S}}(N) = \begin{cases} \frac{1}{N} + \beta\delta\frac{N-1}{N}V^{\mathcal{S}}(N-1) & \textit{if } (m^{\mathcal{S}}, n^{\mathcal{S}}) = (1, N-1), \\ \beta\delta\{\frac{m^{\mathcal{S}}}{N}V^{\mathcal{S}}(m^{\mathcal{S}}) + \frac{n^{\mathcal{S}}}{M}V^{\mathcal{S}}(n^{\mathcal{S}})\} & \textit{otherwise.} \end{cases}$$

This lemma can be used to check the optimality of the strategy $\mathcal{S}$ induced policy $(m^{\mathcal{S}}, n^{\mathcal{S}})$ for the agent's state-N self, and hence the optimality of the strategy $\mathcal{S}$ for the present-biased agent.

Definition 5. The strategy $\mathcal{S}$ induced policy $(m^{\mathcal{S}}, n^{\mathcal{S}})$ in state N is optimal for the agent's state-N self if it is not optimal for the agent's state-N self to deviate to other policies given that he will follow the strategy $\mathcal{S}$ in the future.

Then, the optimality of the strategy $\mathcal{S}$ is defined in the following way.

Definition 6. The strategy $\mathcal{S}$ is optimal for the present-biased agent, if, for all N, the strategy $\mathcal{S}$ induced policy $(m^{\mathcal{S}}, n^{\mathcal{S}})$ in state N is optimal for the agent's state-N self.

The optimality of a strategy can be shown using the same one-step deviation principle as in the benchmark case. The difference is that in the benchmark case, since the agent is time-consistent, it can be considered as the agent's state-N selves are the same for all N. To show the optimality of the Linear Search Strategy (Binary Search Strategy, resp), it is thus sufficient to show that no Linear Search Deviating Strategy (Binary Search Deviating Strategy, resp) is beneficial in an arbitrary state N. However, when the agent is present-biased, the agent's preferences are different in each state N, to show the optimality of a strategy, we need to check that for all N, the agent's state-N self does not want to deviate. This idea coincides with the backward induction. The agent's state-N self chooses the optimal policy in state N given that for all $m < N$, his state-m selves use the optimal policies in state m.

I first show the first bullet point of Proposition 4. I ask the question: if the present-biased agent believes that his future selves will use the Linear Search Policy, what is the smallest state $\underline{N}$ such that the agent's state-$\underline{N}$ self finds it beneficial to use other policies in state $\underline{N}$? If the state $\underline{N}$ does not exist, then I have shown that the first bullet point of Proposition 4 is true. This proof uses the idea of backward induction. Since the agent does not make effective decisions in state $N \leq 3$, where the Linear Search Policy the Binary Search Policy coincide, it can be considered as the agent uses Linear Search Policy in state $N \leq 3$. By backward induction, the optimal policy in state 4 gives the agent's state-4 self the highest value function given that the agent's future selves will use the Linear Search Policy. In state 4, given that the agent's future selves will use the Linear Search Policy, the agent's state-4 self's value function associated with Linear Search is $\mathcal{U}^{L}(4) = \frac{1}{4} + \delta\beta\frac{3}{4}V^{L}(3)$, where $V^{L}(\cdot)$ is the value function associated with the Linear Search Strategy when the agent is time consistent (see Lemma 1). If the agent's state-4 self uses some other policies, which can only be Binary Search in state 4, given the agent's future selves use the Linear Search Policy, then the value function $\mathcal{U}^{D}(\cdot)$ associated with this deviation is $\mathcal{U}^{D}(4) = \beta\delta V^{L}(2)$. By deviating to the Binary Search Policy in state 4, the agent gives up the payment of $\frac{1}{4}$ at present, and increases the discounted continuation value by $\delta[V^{L}(2) - \frac{3}{4}V^{L}(3)]$. Proposition 1 implies that when the agent is time consistent and has the discount parameter $\delta < \bar{\delta}$, the increasing of the discounted continuation value is smaller than the payment the agent gives up today, and the Linear Search Policy is hence optimal in state 4. When the agent is present-biased, he perceives the increasing of the discounted continuation value to be even smaller than the real discounted continuation value. The Linear Search Policy is thus also optimal in state 4 for the present-biased agent's state-4 self. This argument can be generalised to the present-biased agent's state-N self for all N. The Linear Search Strategy is thus optimal for the present-biased agent with $\delta < \bar{\delta}$ and $\beta \in (0,1)$.

To find the optimal strategy of the present-biased agent with $\delta > \bar{\delta}$, I ask the following two questions:

- If the present-biased agent's future selves will use the Binary Search Policy, what is the smallest state $\underline{N}$ such that the agent's state-$\underline{N}$ self finds it beneficial to use other policies in state $\underline{N}$?
- If the present-biased agent's future selves will use the Linear Search Policy, what is the smallest state $\underline{N}$ such that the agent's state-$\underline{N}$ self finds it beneficial to use other policies in state $\underline{N}$?

Answering the first question is useful to show the second bullet point of Proposition 4, while answering the second question is useful to show the third bullet point. If the smallest state $\underline{N}$ in the first question does not exist, then the Binary Search Strategy is optimal.

The proof uses the idea of backward induction. Since the agent does not make any effective decisions in state $N \leq 3$, and the Binary Search Policy and the Linear Search Policy coincide, it can be considered as the agent uses Binary Search Policy in state $N \leq 3$. The optimal policy in state 4 gives the agent's state-4 self the highest value function given that the agent's future selves will use the Binary Search Policy. In state 4, given that the agent's future selves will use the Binary Search Policy, the agent's state-4 self's value function associated with Binary Search is $\mathcal{U}^B(4) = \beta\delta V^L(2) = \beta V^B(4)$, where $V^B(\cdot)$ is the value function associated with the Binary Search Strategy when the agent is time consistent (see Lemma 1). Proposition 1 implies that deviating to any policy that gives the agent zero payment today is not beneficial. Therefore, only the Linear Search Policy should be considered. If the agent's state-4 self believes that his future selves will use the Binary Search Policy, and he uses Linear Search Policy in state 4, the value function is $\mathcal{U}^D(4) = \frac{1}{4} + \beta\delta\frac{3}{4}V^B(3)$. It can be regarded as the agent asks himself this question: in comparison with always using Binary Search Policy, is it beneficial for me to postpone the Binary Search to tomorrow and use Linear Search today? The benefit of using the Linear Search Policy today is the positive expected payment $\frac{1}{4}$, and the cost of using the Linear Search Policy is from the delay of Binary Search, which is $V^B(4) - \delta\frac{3}{4}V^B(3)$. Since this cost is future cost, the present-biased agent perceives the cost as $\beta[V^B(4) - \delta\frac{3}{4}V^B(3)]$. In state 4, delaying the Binary Search to tomorrow is not beneficial when the agent's state-4 self's perceived cost is greater than the benefit, that is, when $\beta > \bar{\beta}^4$, where $\bar{\beta}^4 = \frac{\frac{1}{4}}{V^B(4) - \delta\frac{3}{4}V^B(3)}$ is the ratio of the benefit to the cost of delaying the Binary Search Policy and use the Linear Search Policy in state 4 instead. The discussion above shows that when the present-biased agent has $\delta > \bar{\delta}$ and $\beta > \bar{\beta}^4$, the Binary Search Policy is optimal in state 4 for the agent's state-4 self. Using the idea of backward induction, the Binary Search Policy can be shown to be optimal in state 5 for the agent's state-5 self if $\beta > \bar{\beta}^5$. The Binary Search Policy is thus optimal for the present-biased agent in all the states up to N if $\beta > \max\{\bar{\beta}^m\}_{m=\{4,5,\dots,N\}}$, where $\bar{\beta}^m := \frac{\frac{1}{m}}{V^B(m)-\delta\frac{m-1}{m}V^B(m-1)}$ is the ratio of the benefit to the cost of using the Linear Search Policy in state m given that the agent's future selves will use the Binary Search Policy. The cost $V^B(m) - \delta\frac{m-1}{m}V^B(m-1)$ is increasing in m because the value function $V^B(\cdot)$ is concave, and the benefit $\frac{1}{m}$ is decreasing in m. The value of $\bar{\beta}^m$ is thus decreasing in m. Therefore, if $\beta > \bar{\beta}^4$, then the value of β is greater than $\bar{\beta}^N$ for all $N > 4$. The intuition is that delaying the Binary Search Policy to tomorrow is the most beneficial for the present-biased agent's state-4 self. If the present-biased agent's state-4 self finds it optimal to use the Binary Search Policy, then all the present-biased agent's selves will find it optimal to use the Binary Search Policy. When the present-biased agent is not too present-biased, that is, when β is big enough, the present-biased agent has the same optimal strategy as the time-consistent agent.

In state N, if the agent's future selves all use the Linear Search Policy, and the agent's state-N self uses the $(m,n) \in \mathcal{F}^\dagger$ policy, the highest payoff from these policies is

$$\beta P(N) \equiv \beta \max_{\{m,n\}\in\mathcal{F}^\dagger} \delta\left\{\frac{m}{N}V^L(m) + \frac{n}{N}V^L(n)\right\}$$

where $V^L(\cdot)$ is value associated with the Linear Search Strategy (see Lemma 1). When $\delta > \bar{\delta}$, it has been shown that the maximum value is achieved at the Binary Search Policy in state N. Following the backward induction method, consider the agent's state-4 self. Since Linear Search and Binary Search coincide in states smaller than 4, it can be considered as the agent's future selves use Linear Search Policy in each future states. If the agent uses the Linear Search Policy in state 4, then the expected payoff is $\frac{1}{4} + \beta\delta\frac{3}{4}V^L(3)$. If the agent uses the Binary Search Policy in state 4, then the expected payoff is $\beta P(4)$. Then, given that the agent's future selves all use the Linear Search Policy, the agent's state-4 self uses the Binary Search Policy if $\beta \geq \tilde{\beta}^4$ where $\tilde{\beta}^4 \equiv \frac{\frac{1}{4}}{P(4)-\delta\frac{3}{4}V^L(3)}$ is the ratio of the benefit to the cost of using the Linear Search Policy in state 4. Let $\tilde{\beta}^N \equiv \frac{\frac{1}{N}}{P(N)-\delta\frac{N-1}{N}V^L(N)}$ be the ratio of the benefit to the cost of using the Linear Search Policy in state N. Note that $\tilde{\beta}^4 = \bar{\beta}^4$ because Linear Search and Binary Search coincide in states smaller than 4.

When $\beta < \tilde{\beta}^4$, the agent uses the Linear Search Policy in state 4 given that his future selves also use Linear Search Policy. Then, consider the optimal policy in state 5 given that the agent's all future selves use the Linear Search Policy. Following the same calculation as in state 4, the agent uses the Binary Search Policy if $\beta \geq \tilde{\beta}^5$, and uses the Linear Search Policy if $\beta < \tilde{\beta}^5$.

Lemma 9. *The ratio of the benefit to the cost of using the Linear Search Policy in state N given that the agent's future selves all use the Linear Search Policy $\tilde{\beta}^N$ is decreasing in N.*

This is because the benefit of using the Linear Search Policy in state N is decreasing in N, and due to the concavity of the function $V^L(\cdot)$, the cost of using the Linear Search Policy is increasing in N. As a result, $\tilde{\beta}^N$ is decreasing in N.

Given this lemma, in state 5, it can be concluded that if the agent has the present-biased parameter $\beta \in [\tilde{\beta}^5, \tilde{\beta}^4)$, it is optimal for him to use the Binary Search Policy in state 5, and to use the Linear Search Policy in all future states. For $\beta < \tilde{\beta}^5$, I can keep discussing the agent's policy in state 6 using the same approach. Because of the decreasing property of $\tilde{\beta}^N$, it will be the case that if the agent has the present-biased parameter $\beta \in [\tilde{\beta}^N, \tilde{\beta}^{N-1})$, it is optimal for him to use the Binary Search Policy in state N, and to use the Linear Search Policy in all future states.

B.7 The proof of Proposition 5

Suppose all the players use the Linear Search Strategy. In state N, prior to knowing whether she is active or not, if player i uses the Linear Search Strategy when she is active, her value is

$$V_i^L(N) = p_i \left\{ \frac{1}{N} + \delta \frac{N-1}{N} V_i^L(N-1) \right\} + (1-p_i) \left\{ \delta \frac{N-1}{N} V_i^L(N-1) \right\}.$$

Given the initial condition $V_i^L(2) = p_i$, it can be computed that

$$V_i^L(N) = \frac{1}{N} p_i \left[\frac{1-\delta^{N-1}}{1-\delta} + \delta^{N-2} \right].$$

If player i uses the $(m,n) \in \mathcal{F}^\dagger$ policy in state N, and uses the Linear Search Policy in all other states, prior to knowing whether she is active or not, her value is

$$V_i^D(N) = p_i \left\{ \delta \frac{m}{N} V_i^L(m) + \delta \frac{n}{N} V_i^L(n) \right\} + (1-p_i) \left\{ \delta \frac{N-1}{N} V_i^L(N-1) \right\}.$$

Next, I show that when $\delta \leq \frac{1}{1+p_i}$, we have $V_i^D(N) \leq V_i^L(N)$. That is, given that all other player uses the Linear Search Strategy, it is optimal for player i to use the Linear Search Strategy (if she is active). Let $W_i^L(N) := N V_i^L(N)$ and $W_i^D(N) := N V_i^D(N)$. To show that $V_i^D(N) \leq V_i^L(N)$ is equivalent to show that $W_i^D(N) \leq W_i^L(N)$. We have

$$W_i^D(N) - W_i^L(N) = \delta W_i^L(m) + \delta W_i^L(n) - \delta W_i^L(N-1) - 1.$$

If $\delta > \frac{1}{2}$, $W_i^L(m) + W_i^L(n)$ is maximised at $m = \frac{N}{2}$. If $\delta \leq \frac{1}{2}$, $W_i^L(m) + W_i^L(n)$ is maximised at $m = 2$. When $\delta \leq \frac{1}{2}$,

$$\begin{aligned} W_i^D(N) - W_i^L(N) &< \delta W_i^L(1) + \delta W_i^L(N-1) - \delta W_i^L(N-1) - 1 \\ &= 2\delta p_i \leq p_i < 1. \end{aligned}$$

Therefore, when $\delta \leq \frac{1}{2}$, there exists an equilibrium where all the players use the Linear Search Strategy. When $\delta > \frac{1}{2}$,

$$\begin{aligned} W_i^D(N) - W_i^L(N) &\leq \delta 2 W_i^L(\frac{N}{2}) - \delta W_i^L(N-1) - 1 \\ &= \frac{\delta + (2\delta - 1)\left(\delta^{N-2} - 2\delta^{\frac{N}{2}-1}\right)}{1-\delta} p_i - 1 \\ &< \frac{\delta}{1-\delta} p_i - 1. \end{aligned}$$

If $\delta < \frac{1}{1+p_i}$, we have $W_i^D(N) - W_i^L(N) < 0$. As a result, if $\frac{1}{2} < \delta < \frac{1}{1+p_i}$, given that all the players use the Linear Search Strategy, it is optimal for player i to use the Linear Search Strategy.

B.8 The proof of Proposition 6

In this proof, I first derive the lifetime utility associated with the Linear Search Strategy and the Binary Search Strategy. Next, I show that there is always a Linear Search Deviating Strategy that gives a higher lifetime utility than the Linear Search Strategy. The Linear Search Strategy is hence always sub-optimal. Then, I show that no Binary Search Deviating Strategy gives a higher lifetime utility than the Binary Search Strategy, and hence the Binary Search Strategy is optimal.

To derive the lifetime utility associated with the Linear Search Strategy, I first write down the Bellman equation

$$\mathcal{S}^L(N) = \frac{1}{N} + \frac{N-1}{N}(\mathcal{S}^L(N-1) - c).$$

Iterate backwards and plug in the initial condition $\mathcal{S}(2) = 1$, we have

$$\begin{aligned}\mathcal{S}^L(N) &= \frac{1}{N} + \frac{N-1}{N}\left(\frac{1}{N-1} + \frac{N-2}{N-1}\big(\mathcal{S}^L(N-2) - c\big) - c\right) \\ &= 1 - \frac{(N+1)(N-2)}{2N}c\end{aligned}$$

By using the Binary Search Strategy, the agent learns the state after $\lceil \log_2(N) \rceil - 1$ periods with probability $\frac{2N-2^{\lfloor \log_2 N \rfloor + 1}}{N}$ and learns the state after $\lfloor \log_2(N) \rfloor - 1$ periods with probability $\frac{2^{\lfloor \log_2 N \rfloor + 1} - N}{N}$. The lifetime utility associated with the Binary Search Strategy is hence

$$\begin{aligned}\mathcal{S}^B(N) &= \frac{2N - 2^{\lfloor \log_2 N \rfloor + 1}}{N}\big(1 - (\lceil \log_2(N) \rceil - 1)c\big) + \frac{2^{\lfloor \log_2 N \rfloor + 1} - N}{N}\big(1 - (\lfloor \log_2(N) \rfloor - 1)c\big) \\ &= 1 - \frac{(2N - 2^{\lfloor \log_2 N \rfloor + 1})(\lceil \log_2 N \rceil - 1) + (2^{\lfloor \log_2 N \rfloor + 1} - N)(\lfloor \log_2 N \rfloor - 1)}{N}c.\end{aligned}$$

Next, I show that there is always a Linear Search Deviating Strategy that gives a higher lifetime utility than the Linear Search Strategy if $c > 0$. The Linear Search Deviating Strategy is to choose $(m, n) \in \mathcal{F}^\dagger$ in state N, and use the Linear Search Policy in all other states. Let $\mathcal{S}^D(N)$ be the lifetime utility associated with the Linear Search Deviating Strategy in state N, then

$$\begin{aligned}\mathcal{S}^D(N) &= \frac{m}{N}\mathcal{S}^L(m) + \frac{n}{N}\mathcal{S}^L(n) - c \\ &= 1 - \frac{m^2 + n^2 + N - 4}{2N}c\end{aligned}$$

If N is even, $\mathcal{S}^D(N)$ is maximised at $m = \frac{N}{2}$, where

$$\max \mathcal{S}^D(N) = 1 - \frac{N^2 + 2N - 8}{4N}c > \mathcal{S}^L(N)$$

if $N > 2$. If N is odd, $\mathcal{S}^D(N)$ is maximised at $m = \frac{N-1}{2}$, where

$$\max \mathcal{S}^D(N) = 1 - \frac{N^2 + 2N - 7}{4N}c > \mathcal{S}^L(N)$$

if $N > 3$. Therefore, whenever the Binary Search Strategy and the Linear Search Strategy does not coincide, there is always a Linear Search Deviating Strategy that gives a higher lifetime utility than the Linear Search Strategy if $c > 0$. Linear Search is hence sub-optimal.

Then, I show that no Binary Search Deviating Strategy gives a higher lifetime utility than the Binary Search Strategy. I first derive the expression of $m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1)$, which will be useful for the rest of the proof. I will show that one-step deviation to the Linear Search Policy in state N is not profitable, and then I show that one-step deviation to $(m, n) \in \mathcal{F}^\dagger$ in state N is not profitable. Lastly, I show that search happens in state N when the cost c is smaller than a threshold $\bar{c}(N)$.

Let $K \in \mathbb{Z}^+$ and $K \geq 2$ be a constant.

Lemma 10. *If* $m = 2^K$,

$$m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1) = 1 - Kc.$$

If $m \in [2^K + 1, 2^{K+1} - 1] \cap \mathbb{Z}^+$,

$$m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1) = 1 - (K+1)c.$$

Proof. We have

$$m\mathcal{S}^B(m) = m - (2m - 2^{\lfloor \log_2 m \rfloor + 1})(\lceil \log_2 m \rceil - 1) + (2^{\lfloor \log_2 m \rfloor + 1} - m)(\lfloor \log_2 m \rfloor - 1)c.$$

Then, the difference is

$$\begin{aligned} m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1) = 1 - \Big[& m\big(2(\lceil \log_2 m \rceil - 1) - \lfloor \log_2 m \rfloor - 1\big) \\ & - (m-1)\big(2(\lceil \log_2 (m-1) \rceil - 1) - \lfloor \log_2 (m-1) \rfloor - 1\big) \\ & - 2^{\lfloor \log_2 m \rfloor + 1}\big(\lceil \log_2 m \rceil - \lfloor \log_2 m \rfloor\big) \\ & + 2^{\lfloor \log_2 (m-1) \rfloor + 1}\big(\lceil \log_2 (m-1) \rceil - \lfloor \log_2 (m-1) \rfloor\big)\Big] c \end{aligned}$$

Case 1. First consider the case that $m = 2^K$. In this case, $\lceil \log_2 m \rceil = \lfloor \log_2 m \rfloor = \lceil \log_2 (m-1) \rceil = K$, and $\lfloor \log_2 (m-1) \rfloor = K - 1$. Then,

$$m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1) = 1 - Kc.$$

Case 2. Then consider the case that $m = 2^K + 1$. In this case, $\lceil \log_2 m \rceil = K + 1$, and $\lfloor \log_2 m \rfloor = \lceil \log_2 (m-1) \rceil = \lfloor \log_2 (m-1) \rfloor = K$. Then,

$$m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1) = 1 - (K+1)c.$$

Case 3. Finally, consider the case that $m \in [2^K + 2, 2^{K+1} - 1] \cap \mathbb{Z}^+$. In this case, $\lceil \log_2 m \rceil = \lceil \log_2 (m-1) \rceil = K + 1 = K + 1$, and $\lfloor \log_2 m \rfloor = \lfloor \log_2 (m-1) \rfloor = K$. Then,

$$m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1) = 1 - (K+1)c.$$

$\square$

Next, I show that one-step deviation to the Linear Search Policy in state N is not profitable. Let $D^L(N)$ be the lifetime utility in state N given that the agent uses the Binary Search Deviating Strategy and chooses the Linear Search Policy in state N, We have

$$D^L(N) = \frac{1}{N} + \frac{N-1}{N}\Big(\mathcal{S}^B(N-1) - c\Big).$$

To show that the one-step deviation is not profitable is equivalent to show $ND^L(N) < N\mathcal{S}^B(N)$, which is equivalent to show $1-(N-1)c < N\mathcal{S}^B(N) - (N-1)\mathcal{S}^B(N-1)$. According to Lemma 10, the inequality holds when $N > 2$.

Then, I show that one-step deviation to $(m,n) \in \mathcal{F}^\dagger$ in state N is not profitable. Let $D^P(N)$ be the lifetime utility in state N if the agent uses the Binary Search Deviating Strategy and chooses $(m,n) \in \mathcal{F}^\dagger$ in state N, then

$$D^P(N) = \max_{(m,n)\in\mathcal{F}^\dagger}\{\frac{m}{N}\mathcal{S}^B(m) + \frac{n}{N}\mathcal{S}^B(n) - c\}$$

If I can show that $D^P(N) = \mathcal{S}^B(N)$ when the agent chooses Binary Search in state N, then there is no profitable Binary Search Deviating Strategy. To show this, I introduce the following corollary.

Corollary 1. *For $(m,n) \in \mathcal{F}^\dagger$, the following inequality holds.*

$$m\mathcal{S}^B(m) - (m-1)\mathcal{S}^B(m-1) \geq (n+1)\mathcal{S}^B(n+1) - n\mathcal{S}^B(n)$$

Proof. Given Lemma 10, I consider four cases. Let $K, J \in \mathbb{Z}^+$ and $K, J \geq 2$ be two constants. **Case 1.** First consider the case that $m = 2^K$ and $n+1 = 2^J$. The left-hand side of the inequality is $1-Kc$ and the right-hand side of the inequality is $1-Jc$. Since $m \leq n$, we have $K < J$. Therefore, the inequality holds with the strict inequality. **Case 2.** Next, consider the case that $m = 2^K$ and $n+1 \in [2^J+1, 2^J-1]$. The left-hand side of the inequality is $1-Kc$ and the right-hand side of the inequality is $1-(J+1)c$. Since $m \leq n$, we have $K < J$. Therefore, the inequality holds with the strict inequality. **Case 3.** Next, consider the case that $m \in [2^K+1, 2^K-1]$ and $n+1 = 2^J$. The left-hand side of the inequality is $1-(K+1)c$ and the right-hand side of the inequality is $1-Jc$. Since $m \leq n$, we have $m < n+1$ and hence $2^{K+1} < 2^J+1$. Since $m, n \in \mathbb{Z}^+$, it must be that $K+1 \leq J$. Therefore, the inequality holds with the weak inequality. **Case 4.** Lastly, consider the case that $m \in [2^K+1, 2^K-1]$ and $n+1 \in [2^J+1, 2^J-1]$. The left-hand side of the inequality is $1-(K+1)c$ and the right-hand side of the inequality is $1-(J+1)c$. Since $m \leq n$, we have $K \leq J$. Therefore, the inequality holds with the weak inequality. $\square$

Given Corollary 1, the following inequality holds

$$m\mathcal{S}^B(m) + n\mathcal{S}^B(n) \geq (m-1)\mathcal{S}^B(m-1) + (n+1)\mathcal{S}^B(n+1).$$

Therefore, $D^P(N)$ is achieved by using the Binary Search Policy in state N. Therefore, there is no profitable Binary Search Deviating Strategy.

Since the agent only searches when learning gives the agent a non-negative payoff. That is $V^B(N) \geq 0$. Therefore, the fixed cost c has to be smaller than the threshold $\bar{c}(N)$ so that the agent starts searching.

B.9 The proof of Proposition 8

This proof follows the following steps. I first write down the Bellman equation. Next, I compute the value associated with the Focal Point Search Strategy. Then, I derive a sufficient condition under which there is no one-step deviation strategy that gives the agent a higher payoff than the Focal Point Search Strategy.

The Bellman equation consists of the payoff at the current period and the continuation value. The agent's revised belief determines the continuation value. If the prior belief is a distribution with a peak, the revised belief at the next time can be one of the following three distributions: a degenerated distribution, a uniform distribution, or a distribution with a peak with a different support. The shape of the revised belief depends on the agent's policy at that time. If the agent uses the Linear Search Policy, the revised belief will be a degenerated distribution or a uniform distribution. If the agent chooses $(m,n) \in \mathcal{F}^\dagger$, the revised belief will be a uniform distribution or a distribution with a peak with a different support. Since the degenerated revised belief means that the agent learns the unknown parameter, the continuation value is hence zero. The positive continuation value thus takes two different functional forms: one corresponding to the uniform revised belief, and the other corresponding to the belief with a peak.

Let $V_p(N)$ be the value function in state N when the belief of the agent is a distribution with a peak of f_1. Let $V_u(N)$ be the value function in state N when the belief of the agent is a uniform distribution. If the agent's belief in state N is the distribution with a peak f_1, the Bellman equation in state N is

$$V_p(N) = \max\left\{ f_1 + (1-f_1)\delta V_u(N-1), \max_{(m,n)\in\mathcal{F}^\dagger} \delta\big\{\mu V_p(m) + (1-\mu)V_u(n)\big\}\right\},$$

with the initial condition $V_p(1) = \frac{1}{\delta}$ and $\mu = f_1 + \frac{1-f_1}{N-1}(m-1)$. The first element is the value associated with the Linear Search Policy in state N, and the second element without the max operator is the value associated with (m,n) in state N. With the max operator, it is the highest value the agent can get by choosing $(m,n) \in \mathcal{F}^\dagger$.

Let $V_p^F(N)$ be the value associated with the Focal Point Search Strategy. Then,

$$V_p^F(N) = f_1 + (1-f_1)\delta V^B(N-1),$$

where $V^B(\cdot)$ is the value function in the benchmark case associated with the Binary Search Strategy (see Lemma 1). This is because if the agent uses the Focal Point Search Strategy and does not learn the unknown parameter, the revised belief becomes the uniform distribution. Then the problem becomes the one that has been discussed in the benchmark case. Since the discount parameter is greater than a half, the corresponding value function is $V^B(\cdot)$.

To check the optimality of the Focal Search Strategy, I derive a sufficient condition under which there is no one-step deviation strategy that gives the agent a higher payoff than the Focal Point Search Strategy.

Definition 7. The Focal Point Search Deviating Strategy is an one-step deviation strategy such that the agent chooses $(m, N-m) \in \mathcal{F}^\dagger$ in state N and uses the Focal Point Search Strategy in all other states.

Let $G(m)$ be the value associated with the Focal Point Search Deviating Strategy. Then,

$$G(m) = \delta\Big\{\mu V_p^F(m) + (1-\mu)V^B(N-m)\Big\}.$$

Plug in the value function associated with the Focal Point Search Strategy, we have

$$G(m) = \delta\Big\{\mu\Big(f_1 + (1-f_1)\delta V^B(m-1)\Big) + (1-\mu)V^B(N-m)\Big\}.$$

The functional form of $V^B(\cdot)$ is known (see Lemma 1). Since $m \geq 2$ and $m \leq N-m$, the upperbound of $V^B(m-1)$ is $V^B(1)$, which is $\frac{1}{\delta}$, and the upperbound of $V^B(N-m)$ is $V^B(2) = 1$. Therefore,

$$G(m) \leq \delta\Big\{\mu\Big(f_1 + (1-f_1)\delta\frac{1}{\delta}\Big) + (1-\mu)\Big\} = \delta.$$

Then, if $\delta \leq f_1$, it is always true that $G(m) \leq V_p^F(N)$. As a consequence, when $\delta \in (\frac{1}{2}, f_1]$, there is no Focal Point Search Deviating Strategy that gives the agent a higher payoff than the Focal Point Search Strategy. The Focal Point Search Strategy is hence optimal.

References

Callander, S. (2011), 'Searching and learning by trial and error', *American Economic Review* **101**(6), 2277–2308.

Deb, R. & Stewart, C. (2018), 'Optimal adaptive testing: Informativeness and incentives', *Theoretical Economics* **13**(3), 1233–1274.

Dejarnette, P., Dillenberger, D., Gottlieb, D. & Ortoleva, P. (2018), Time Lotteries and Stochastic Impatience.

Dillenberger, D. (2010), 'Preferences for One-Shot Resolution of Uncertainty and Allais-Type Behavior, Second Version', *Econometrica* **78**(6), 1973–2004.

Dillenberger, D., Gottlieb, D. & Ortoleva, P. (2018), Stochastic Impatience and the Separation of Time and Risk Preferences.

Epstein, L. G. & Zin, S. E. (1989), 'Substitution, Risk Aversion, and the Temporal Behavior of Consumption and Asset Returns: A Theoretical Framework', *Econometrica* **57**(4), 937.

Fischer, C. (1999), Read This Paper Even Later: Procrastination with Time-Inconsistent Preferences.

Kreps, D. M. & Porteus, E. L. (1978), 'Temporal Resolution of Uncertainty and Dynamic Choice Theory', *Econometrica* **46**(1), 185.

Laibson, D. (1997), 'Golden Eggs and Hyperbolic Discounting', *The Quarterly Journal of Economics* .

Meyer, M. A. (1994), 'The Dynamics of Learning with Team Production: Implications for Task Assignment', *The Quarterly Journal of Economics* **109**(4), 1157–1184.

O'Donoghue, T. & Rabin, M. (1999), 'Doing It Now or Later', *American Economic Review* **89**(1), 103–124.

Palacios-Huerta, I. (1999), 'The Aversion to the Sequential Resolution of Uncertainty', *Journal of Risk and Uncertainty* **18**(3), 249–269.

Phelps, E. S. & Pollak, R. A. (1968), 'On second-best national saving and game-equilibrium growth', *The Review of Economic Studies* **35**(2), 185–199.

Zhong, W. (2017), Time preference and information acquisition.

Zhong, W. (2019), Optimal Dynamic Information Acquisition.