Stochastic Model for Sunk Cost Bias [*]

Jon Kleinberg[1]   Sigal Oren[2]   Manish Raghavan[1]   Nadav Sklar[2]

[1]Computer Science Dept., Cornell University, Ithaca, New York, USA
[2]Computer Science Dept., Ben-Gurion University of the Negev, Be'er Sheva, Israel

Abstract

We present a novel model for capturing the behavior of an agent exhibiting sunk-cost bias in a stochastic environment. Agents exhibiting sunk-cost bias take into account the effort they have already spent on an endeavor when they evaluate whether to continue or abandon it. We model planning tasks in which an agent with this type of bias tries to reach a designated goal. Our model structures this problem as a type of Markov decision process: loosely speaking, the agent traverses a directed acyclic graph with probabilistic transitions, paying costs for its actions as it tries to reach a target node containing a specified reward. The agent's sunk cost bias is modeled by a cost that it incurs for abandoning the traversal: if the agent decides to stop traversing the graph, it incurs a cost of $\lambda \cdot C_{sunk}$, where $\lambda \geq 0$ is a parameter that captures the extent of the bias and C_{sunk} is the sum of costs already invested.

We analyze the behavior of two types of agents: naive agents that are unaware of their bias, and sophisticated agents that are aware of it. Since optimal (bias-free) behavior in this problem can involve abandoning the traversal before reaching the goal, the bias exhibited by these types of agents can result in sub-optimal behavior by shifting their decisions about abandonment. We show that in contrast to optimal agents, it is computationally hard to compute the optimal policy for a sophisticated agent. Our main results quantify the loss exhibited by these two types of agents with respect to an optimal agent. We present both general and topology-specific bounds.

1 INTRODUCTION

Imagine that you paid \$50 to go to a rock concert and five minutes into the show you realize that the acoustics are horrible, the venue is smelly and the band is not playing well. Will you stay or go? Would you have made a different decision if the concert were free? Many will choose to stay in the concert in the first case but leave in the second one. This phenomenon, in which effort or cost invested in the past affects current decisions, has fascinated many researchers from different disciplines. This is evident from the variety of names the phenomenon has been studied under: the sunk cost effect [Arkes and Blumer, 1985, Thaler, 1980], escalation of commitment [Staw, 1976] and the Concorde fallacy [Dawkin, 1976, Weatherhead, 1979]. The latter is named after the famous supersonic airplane whose development was continued long after it was clear that it did not have any economic justification. Some of the many situations in which sunk cost has been observed include auctions [Augenblick, 2016], medical treatment [Coleman, 2010, Eisenberg et al., 2012], project development [Garland, 1990] and poker [Smith et al., 2009].

Factoring sunk cost into future decisions is at odds with standard economic theory advocating that decisions should only depend on marginal costs and gains. Several explanations have been offered for the sunk cost effect. Arkes and Blumer [1985] suggest it is a manifestation of the "do not waste" rule that we are often taught as children. Early work in psychology [Aronson, 1968, Staw, 1980] attributes this to self justification: decision-makers continue in the same course of action to justify their initial decision and avoid cognitive dissonance. Thaler [1980] applies prospect theory [Kahneman and Tversky, 1979] to explain this bias.

In this paper, we analyze the performance of agents who engage in activities that require multi-step planning in the presence of sunk cost bias. Through this, our work is situated in a recently growing literature in algorithmic game theory aiming to model and theoretically analyze planning related biases (e.g., [Kleinberg and Oren, 2014, Gravin et al.,

---

2016, Kleinberg et al., 2016, Albers and Kraft, 2016, Tang et al., 2017, Kleinberg et al., 2017]). Despite the crucial role played by sunk cost bias in empirical studies of behavior, it has received very little theoretical study in this style; the main prior contribution is a model of Kleinberg et al. [2017] that considered the interplay of sunk cost bias with present bias in a deterministic setting. However, looking at the scenarios discussed so far, we see that many of them crucially involve agents who are planning with respect to uncertainty about future outcomes: the sunk cost bias often becomes particularly dangerous when an agent takes an action while the future remains uncertain, and then is subject to the sunk cost from this action after the uncertainty is resolved. Indeed, many of the most natural questions that arise when studying sunk cost bias in isolation (separately from other effects such as present bias) do not have natural formulations in deterministic models. It is therefore an important and unexplored question to analyze the effects of sunk cost bias in a model featuring uncertainty.

A Stochastic Model of Sunk Cost. We present a stochastic model aiming to study scenarios involving sunk cost bias and uncertainty. We focus on situations in which taking sunk cost into account is irrational,[1] for example, a gambler who has already "invested" $100 in a slot machine and keeps playing because they are sure that after "investing" so much money they will hit the jackpot soon. These situations involve the following basic ingredients: an agent needs to formulate a plan in which it traverses a set of states, trying to reach a designated goal state. The transitions between states are stochastic based on the agents' actions, and the agent must deal with its own sunk cost bias as it formulates and updates its plan for traversing the states.

Motivated by these considerations, we model the agent's problem using a directed acyclic graph in which the agent must traverse a path from a start node s to a target node t, with a reward of R for reaching t. Each node is assigned a cost of going forward and there is a probability distribution on its outgoing edges, determining the next node that the agent would reach. This is a type of a Markov decision process. After each step, the agent has to choose whether to stop or keep traversing the graph. If the agent at some node v decides to continue traversing the graph then it pays the cost assigned to v and moves to a neighboring node of v determined stochastically according to a distribution on v's neighbors.

We model the decision making process of agents with sunk cost bias similarly to Kleinberg et al. [2017]. We assume that an agent exhibiting sunk cost bias has some parameter $\lambda \geq 0$ that represents the extent to which the agent cares

about sunk cost[2]. While the regime of $0 \leq \lambda \leq 1$ is perhaps more natural, for sake of generalization, we present and analyze our model using the more general assumption of $\lambda \geq 0$. Let C_{sunk} be the cost that the agent already invested. An agent with sunk cost bias views the option of quitting as having a cost of λC_{sunk}, hence it will continue traversing the graph if and only if the expected payoff from continuing is greater than $-\lambda C_{sunk}$. We stress that, while our paper uses the basic means of accounting for sunk cost employed by Kleinberg et al. [2017], the models studied in the two papers are inherently different. The current paper studies a stochastic model focusing on the effects of sunk cost bias by itself; in contrast, the model of Kleinberg et al. [2017] is a deterministic formalism that studies the simultaneous effect of present bias and sunk cost bias, and through its deterministic structure cannot encapsulate the key issues that we address here.

Following O'Donoghue and Rabin O'Donoghue and Rabin [1999, 2001], we analyze the behavior of two types of agents: naive agents that are unaware of their bias, and sophisticated agents that are aware of it. Even though a sophisticated agent is aware of its bias it cannot simply ignore it. However, loosely speaking, it can take future actions to minimize the negative implications of its bias. To understand the behavior of the different agents, it is best to walk through a simple example. Consider a slot machine with a probability of $1/3$ of winning a reward of $10. As part of a promotion the casino prices the first round at $3 instead of the usual price of $4. This scenario is depicted in the graph in Figure 1. An unbiased (e.g., optimal) agent would only play as long as the expected payoff is greater than 0. Hence, in this game, it will only play the first round. Now, consider a naive agent exhibiting sunk cost bias with a parameter $\lambda = 1/3$. If it loses the first round it would have a sunk cost of $1/3 \cdot 3 = 1$. Thus, its payoff for quitting would be -1 while its expected payoff for continuing would be $1/3 \cdot 10 - 4 = -2/3$. The naive agent will therefore play the next round as well and attain a negative expected payoff. In fact, we show that the negative payoff of a naive agent can be exponentially large in the size of the graph.
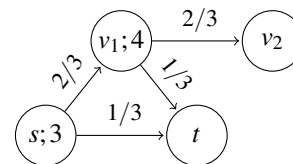

Figure 1: For $R = 10$, naive sunk cost bias agents will continue at v_1 and end up with a negative expected payoff.

A sophisticated agent with sunk cost bias is aware of its bias.

---

[1] We note that as McAfee et al. [2010] advocates taking sunk cost into account can sometimes be rational. For example, in a project with an unknown completion time, the time already invested can hint at the actual completion time.

[2] It is natural to limit λ to non-negative values as negative values imply that the agent believes that if it would stop it would get some of its investment back. This is the opposite of sunk cost bias.

This means that it knows which action it will take in any subsequent node for any possible sunk-cost and can use this information to compute its expected payoff. This is different than a naive agent who is unaware of its bias and hence, wrongfully, believes that in the future, it will behave the same as an unbiased agent and therefore, in all subsequent nodes, will have the same expected payoff as an unbiased agent.

We observe that the expected payoff of a sophisticated agent is always non-negative since it would stop traversing the graph if it knows that its expected payoff for doing so will be negative. This is the case in the example in Figure 1 in which the sophisticated agent knows that if it will play the first round and lose than it will also play the second round. This implies that its expected payoff for playing the slot machine is $1/3 \cdot 10 - 3 + 2/3(1/3 \cdot 10 - 4) = -1/9$. The fact that sophisticated agents sometimes stop traversing the graph prematurely makes their payoff potentially smaller than that of an optimal agent, although they avoid some of the more dramatic payoff shortfalls of naive agents.

Results. We begin by considering naive agents. Since they believe that in the future they will behave as optimal agents, their policy can be efficiently computed similarly to the policy of optimal agents. This is done by going over the graph in reverse topological order and computing the expected payoff of continuing at each node. Naive agents can then decide whether to continue or not by comparing these values against their sunk cost. We show that since they are oblivious to their sunk cost bias, we can construct instances in which they accumulate some sunk cost in the beginning. Then, due to this sunk cost, they continue traversing the graph and accumulate more and more sunk cost even when their expected payoff is negative. As a result, they may end up with a negative payoff that is exponential in the graph's size. This result illustrates the danger of marketing strategies that reduce initial entrance costs to lure individuals to begin some risky endeavor (e.g., as the investing app Robinhood gives a free stock to anyone opening a new account) or take on some bad habit (e.g., tobacco companies giving free cigarettes to employees).

Our main focus in this paper is on sophisticated agents. The behavior of agents that are aware of their bias is much more complex. In contrast to optimal and naive agents, they cannot compute their optimal policy by going over the nodes in reverse topological order. This is because the decision of whether to stop or continue at each node depends on the amount of sunk cost they accumulated along the way. When there are different paths reaching the same node, the amount of sunk cost may vary depending on the realized path. In fact, we show that the problem of computing the optimal policy for a sophisticated agent is #P-Hard. This is done by reducing from the $0-1$ knapsack solution counting problem. Roughly speaking, we construct instances in which computing the expected payoff of a sophisticated

agent if it starts traversing the graph requires counting the number of valid solutions to a corresponding knapsack problem. It is worth noting that a different type of hardness (i.e., NP-hardness) was proven by Kleinberg et al. [2017] for sophisticated agents exhibiting both sunk cost bias and present bias. The results strengthen one another and show that under different models being sophisticated about one's sunk cost bias may be quite challenging. As part of future research, it would be fascinating to model and analyze heuristics that individuals may use to bypass this hardness.

We continue with comparing the payoff of a sophisticated agent against the payoff of an optimal agent. Roughly speaking, sophisticated agents exhibit the opposite problem than naive agents: they take a too conservative approach and stop traversing the graph prematurely. As a result, they can have a payoff of 0 even when the optimal agent has a positive payoff. When λ is approaching infinity this gap can attain its maximal value which is R. However, the payoff difference of R is far from tight for smaller values of λ. Hence, we look for tighter bounds that are more suitable for such values. We provide a number of bounds on the difference between the payoff of optimal and sophisticated agents. For example, we show that $\pi_s \geq \pi_o - \frac{\lambda}{1+\lambda} \cdot R$, where π_s and π_o are the expected payoffs of the sophisticated and optimal agents respectively. We present some evidence that this bound is not tight, particularly, for $0 \leq \lambda \leq 1$. We suspect that graphs achieving the worst case difference, for $0 \leq \lambda \leq 1$, are fan graphs (graphs that include a path plus an edge from each node in the path to the target). We show that for such graphs $\pi_s \geq \pi_o - \frac{1}{e} \cdot \lambda \cdot R$ and prove that this bound is essentially asymptotically tight (in the graph's size).

2 MODEL AND NAIVE AGENTS

In our model an agent is traversing a directed acyclic graph (i.e., DAG). The graph is a Markov decision process (i.e., MDP) where each state u has a cost $c(u)$ [3] which is the cost of an agent at u to continue traversing the graph and for each neighboring states u and v, $p(u,v)$ denotes the probability of a $u \rightarrow v$ transition. The graph also has a designated target node t. If the agent reaches t it receives a reward of R. An agent traversing the graph forms a policy that decides for each node in the graph whether to continue traversing the graph or not. The goal of an agent is to choose a policy that maximizes its *expected payoff* – the probability of reaching the target multiplied by R minus the expected cost. We can define the expected payoff of an agent inductively as follows: if the agent decides to continue from u, the expected payoff at a vertex u is the weighted average of the expected payoff of each neighbor vertex minus the cost for continuing. If it decides to stop, its expected payoff is 0. We denote the expected payoff of an optimal agent (i.e., bias-free agent)

---

[3]This is a restricted type of MDP in which for every node u the transition cost to each neighbor is the same.

currently at node u by $\pi_o(u)$. We have that:

$$\pi_o(u) = \max\{\sum_{v \in N(u)} p(u,v) \cdot \pi_o(v) - c(u), 0\}$$

where $N(u)$ denotes the set of neighbors of node u.

An agent with sunk cost bias is characterized by a parameter $\lambda \geq 0$ that captures the intensity of its bias. We begin by considering naive agents. These agents are unaware of their bias and as a result plan as if they will behave optimally in the future. A naive sunk cost bias agent, perceives the cost of stopping as λ multiplied by the cost it invested. Therefore, it will continue traversing the graph at node u after accumulating a sunk cost of K if and only if $\sum_{v \in N(u)} p(u,v) \cdot \pi_o(v) - c(u) \geq -\lambda K$. For consistency, we use the term expected payoff, for all types of agents to denote their *actual* expected payoff and not the perceived one. Thus, the expected payoff of the naive agent is $\pi_n(u,K) = 0$ if it decides to abandon and is otherwise

$$\pi_n(u,K) = \sum_{v \in N(u)} p(u,v) \cdot \pi_n(v, K + c(u)) - c(u).$$

It is not very hard to construct instances in which the payoff of the naive agent is negative. Here, we show that its negative payoff can grow exponentially in the graph's size. We first get the agent to incur sufficient sunk cost. Once the agent incurred this amount of sunk cost it would rather continue with no chance of reaching any reward than cut its losses and stop. Essentially, the agent prefers continuing a lost cause to admitting defeat. It always believes that it will take just one more step and then give up, but once it gets there, it finds that it no longer wants to give up.

Claim 2.1 *The negative expected payoff of a naive agent may be exponential in the graph's size.*

Proof: Consider the instance depicted in Figure 2. Observe that a naive agent will choose to continue at s since the expected payoff is 0.[4] If it ends up at v_1 it will again choose to continue, for the same reason, for any value of λ. Now for any $2 \leq i \leq n-1$, the agent at v_i will continue to v_{i+1} since it will accumulate a cost of

$$R + \sum_{j=0}^{i-3} \lambda R (1+\lambda)^j = R \cdot (1+\lambda)^{i-2}$$

Thus, it will have a perceived cost of stopping of $\lambda R \cdot (1+\lambda)^{i-2}$ which is the same as the cost it thinks it will have for continuing one step and then stopping. Since the expected payoff of the two first steps is 0, we get that the expected payoff of the naive agent is $-\frac{1}{4}R(1+\lambda)^{n-2}$. $\square$

3 SOPHISTICATED SUNK COST BIAS AGENT

Recall that a sophisticated agent is aware of its bias and hence it can compute its *actual* expected payoff for continuing traversing the graph. Hence, a sophisticated agent at node u that accumulated a sunk cost of K decides to continue if its (actual) expected payoff is at least $-\lambda K$. For this reason, a sophisticated agent will only traverse the graph if its expected payoff is non-negative. Nevertheless, for any W, we can construct an instance in which a sophisticated agent has an expected payoff of 0 while the optimal agent has an expected payoff of $W > 0$. Consider Figure 3, with $R = 10W$ and $\lambda = 0.5$. An optimal agent would continue at s and stop at u, with expected payoff W. However, a sophisticated agent would continue at u as well, so its expected payoff would be 0.

As we discussed, computing a policy for optimal or naive agents is computationally easy. We observe that this is not the case for sophisticated agents. Since a sophisticated agent is aware of its bias, in order to compute its expected payoff it should know which action its future selves will take in each of the following nodes. This problem is complicated by the fact that the choice depends on the amount of sunk cost they accumulated in reaching these nodes. We observe that there are instances in which computing the expected payoff is computationally hard. Hence, it is computationally hard for a sophisticated agent to compute its policy.

Proposition 3.1 *The problem of computing the optimal policy for a sophisticated sunk cost bias agent is #P-hard.*

Proof: We use the following reduction from knapsack solution counting. We are given a capacity for the knapsack B and n objects with weights $w_1, \ldots w_n$. Our goal is to count the number of subsets S such that $\sum_{i \in S} w_i \leq B$. Given an instance of the counting problem we construct the Markov decision process illustrated in Figure 4.

Let K be the cost incurred after s and just before q, meaning K is the sum of some subset of the w_i's. We observe that at q', the agent will have a different policy based on the value of K. Continuing at q' yields a net payoff of $-\lambda(B+C)$, while the perceived cost of stopping is $-\lambda(K+C)$. Hence, the agent continues at q' if and only if $K > B$.[5] Thus, we can separate all paths leading to q into two types based on whether the agent would stop or continue at q' if it reaches it. We first prove correctness under the assumption that the agent will always reach q and then show it holds.

For paths along which the agent would not continue at q', the expected payoff is $\pi_s^{\varnothing} = \frac{R}{2} - \mathbb{E}[K|K \leq B] - C$. This is because such paths reach t when a $q \to t$ transition is realized
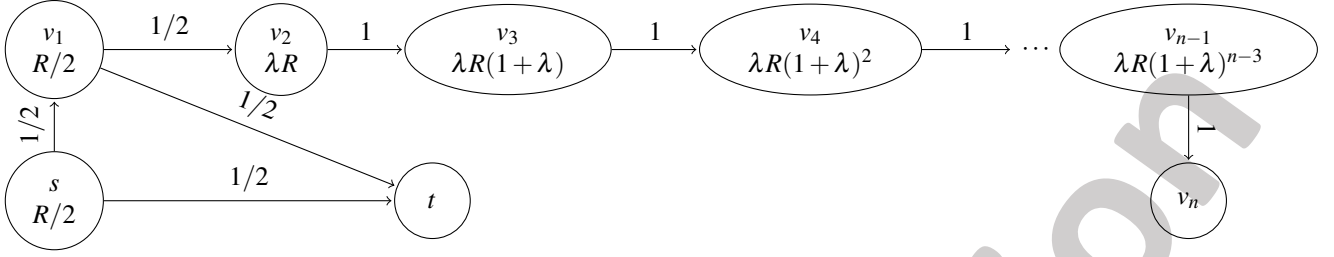
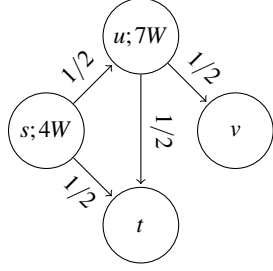Figure 2: Graph for which the expected payoff are exponentially negative.

Figure 3: Sophisticated agent gets 0, optimal agent gets W.

(with probability $\frac{1}{2}$). Note that we can complete the reduction without knowing the value of $\mathbb{E}[K|K \leq B]$. For paths along which the agent would continue at q' the expected payoff is

$$\pi_s^{\copyright} = \frac{R}{2} - \mathbb{E}[K|K > B] - \frac{\lambda}{2}(B+C) - C.$$

This is because these paths always reach t, but they incur a cost of $R + \lambda(B+C)$ with probability $\frac{1}{2}$. Let p be the probability that $K \leq B$. The total expected payoff is then

$$\begin{aligned}
\pi_s &= p \cdot \pi_s^{\oslash} + (1-p) \cdot \pi_s^{\copyright} \\
&= p\frac{R}{2} + (1-p)\frac{R}{2} - p \cdot \mathbb{E}[K|K \leq B] - (1-p) \cdot \mathbb{E}[K|K > B] \\
&\quad - p \cdot C - (1-p) \cdot \left(\frac{\lambda}{2}(B+C) + C\right) \\
&= \frac{R}{2} - \frac{1}{2}\sum_{i=1}^{n} w_i - C - \frac{\lambda}{2}(1-p)(B+C)
\end{aligned}$$

where in the last step we use:

$$p \cdot \mathbb{E}[K|K \leq B] + (1-p) \cdot \mathbb{E}[K|K > B] = \mathbb{E}[K] = \frac{1}{2}\sum_{i=1}^{n} w_i.$$

By rearranging we get that:

$$p = \frac{2\pi_s + \sum_{i=1}^{n} w_i - R + 2C}{\lambda(B+C)} + 1$$

Since the number of paths such that $K \leq B$ is $p \cdot 2^n$, and the number of such paths is the same as the number of solutions to the knapsack problem, we have

$$\# \text{ Solutions} = 2^n \left(\frac{2\pi_s + \sum_{i=1}^{n} w_i - R + 2C}{\lambda(B+C)} + 1\right).$$

Hence, if the sophisticated agent can compute its expected payoff in polynomial time, it can also solve the #P-hard problem of knapsack solution counting. To complete the reduction, we first show that the agent will always reach q if it starts to traverse the graph. To do this, we observe that from any node before q (not including s) the expected payoff is at least $\frac{R}{2} - \lambda(B+C) - \sum_{i=1}^{n} w_i$. Therefore, the agent will continue as long as

$$\frac{R}{2} - \lambda(B+C) - \sum_{i=1}^{n} w_i \geq -\lambda C.$$

By rearranging we get the reduction holds for any $R \geq 2\sum_{i=1}^{n} w_i + 2\lambda B$.

Finally, recall that we need to show that the problem of computing the optimal policy for a sophisticated agents is #P-hard. This requires us to construct an instance in which the agent has to know the exact expected payoff to compute its policy. We prove the following claim:

Claim 3.2 *For any $0 < \alpha < 1$ there exists C and R such that a sophisticated agent will traverse the graph if and only if #Solutions $\geq \alpha \cdot 2^n$.*

Proof: Let $R = 2\sum_{i=1}^{n} w_i + 2\lambda B$. We show that for any $0 < \alpha < 1$ there exists C such that the expected payoff of the sophisticated agent is 0 if $p = \alpha$. This implies that the agent will traverse the graph if and only if $p \geq \alpha$ as required. Observe that

$$\pi_s = \frac{R}{2} - \frac{1}{2}\sum_{i=1}^{n} w_i - C - \frac{\lambda}{2}(1-\alpha)(B+C) = 0$$

$$\implies C = \frac{R - \sum_{i=1}^{n} w_i - \lambda(1-\alpha)B}{2 + \lambda(1-\alpha)}$$

Finally, notice that both R and C have a polynomial representation as they are defined by arithmetic operations over numbers that their representation is polynomial in the problem's size. $\square$

The claim implies the proof of Proposition 3.1 since if the sophisticated agent can compute its optimal policy for any instance constructed for $0 < \alpha < 1$, we can use its poly-time algorithm to run a binary search over the fractions of valid
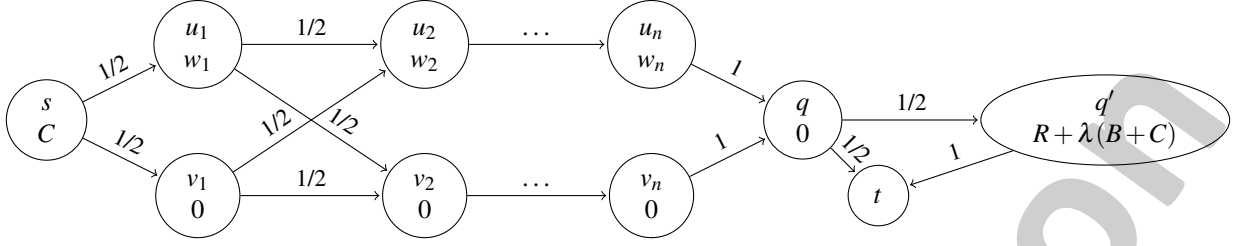
Figure 4: Reduction from knapsack solution counting to computing the payoff of a sophisticated sunk cost biased agent.

solutions. Since the size of the search space is 2^n, our binary search will compute the exact number of solutions to the knapsack problem in polynomial time. □

Note that even though a naive agent can efficiently decide on its actions (as it plans to behave optimally), the same reduction as in the proof of Proposition 3.1 shows that computing the expected payoff of a naive agent is also #P-Hard.

3.1 BOUNDING THE PAYOFF OF A SOPHISTICATED AGENT

In this section, we try to further understand how much smaller the payoff of a sophisticated agent may be relative to the payoff of an optimal agent. As discussed in the example in Figure 3, we cannot have a multiplicative bound here as in some cases the payoff of the optimal agent can be positive while the payoff of a sophisticated agent is 0. As the payoff of an optimal agent is at most R we have that for any value of $\lambda \geq 0$, $\pi_s \geq \pi_o - R$. In fact, this bound is asymptotically tight even for a 3-node graph as λ is approaching infinity. We prove this by showing in the supplementary material that for any value of $\lambda \geq 0$ there exists a 3-node instance such that, $\pi_s = \pi_o - \frac{2+\lambda-2\sqrt{1+\lambda}}{\lambda} \cdot R$. This implies that there exists a family of instances such that as λ is approaching infinity $\pi_o - \pi_s$ is approaching R. While the trivial bound $\pi_s \geq \pi_o - R$ is asymptotically tight for very large values of λ, as we will see next, it is far from tight for smaller values of λ. Hence, for the rest of the paper, we look for tighter bounds on the payoff of sophisticated agents.

To help analyze the payoff of the sophisticated agent we define an auxiliary "hybrid" agent with the following behavior: it continues wherever the optimal agent continues until it either reaches the goal or the optimal agent stops. If the optimal agent stops, then it does what the sophisticated agent would have done. We show that the payoff of the sophisticated agent is always higher than the expected payoff of the "hybrid" agent and then provide bounds on the expected payoff of a hybrid agent.

Proposition 3.3 *The expected payoff of a sophisticated agent is at least the expected payoff of a hybrid agent.*

Proof: We label the nodes of the graph according to the

manner in which the hybrid agents behaves: nodes that it behaves as an optimal agent are labeled by o and those at which it behaves as a sophisticated agent by s. In particular, nodes that the optimal agent reaches and decides to continue are labeled by o and the rest of the nodes are labeled by s. Since the decisions of the optimal agent do not depend on the path leading to a node this is a well defined distinction. Moreover, once the hybrid agent starts behaving as a sophisticated agent it would keep doing so until it either reaches the target or stops. Thus, starting from any s-labeled node the sophisticated and hybrid agents will behave the same.

Denote the payoff of the hybrid agent by π_h. We show that for every o-labeled node u and every K which is the weight of a path from s to u we have that $\pi_h(u,K) \leq \pi_s(u,K)$. Assume towards a contradiction that there exists an o-labeled node u and a path reaching this node of cost K such that $\pi_h(u,K) > \pi_s(u,K)$. If there is more than one such node, let u be the last such node in the topological order. This implies that for every node v subsequent to it and any $K' > 0$ a cost of a path reaching v we have that $\pi_h(v,K') \leq \pi_s(v,K')$.

Since in all o-labeled nodes the optimal agent decides to continue there are two cases. In the first case, at u the sophisticated agent decides to continue as well, thus the expected payoff is the weighted average over all successor nodes, and hence $\pi_h(u,K) \leq \pi_s(u,K)$. In the second case, the sophisticated agent stops and the hybrid agent continues. We know that the expected payoff of the sophisticated agent if it continued would be non-positive, as the sophisticated agent decides to stop, and the expected payoff of the hybrid agent is the weighted average over all successors. Again, by our assumption on subsequent nodes, we have that the weighted average over these successors for the hybrid agent must be no more than the same weighted average for the sophisticated agent, which we know is no more than 0. Thus, $\pi_h(u,K) \leq \pi_s(u,K)$. □

In order to lower-bound the payoff of a sophisticated agent, we now take a closer look at the payoff of an optimal agent. Let S denote the event in which the optimal agent reaches t (starting from s) and $p(S)$ the probability of this event. The payoff of an optimal agent is: $\pi_o = p(S) \cdot R - E[C]$, where C is a random variable equal to the cost that the optimal agent incurred. By decoupling the event in which the optimal

agent reaches the target and does not reach the target:

$$\pi_o = p(S) \cdot R - p(S) \cdot E[C|S] - (1 - p(S)) \cdot E[C|\bar{S}]$$

Based on this we can bound the payoff of the hybrid agent:

Lemma 3.4 $\pi_h \geq \pi_o - \lambda \cdot (1 - p(S)) \cdot E[C|\bar{S}]$

Proof: The hybrid agent reaches the target with probability at least $p(S)$ since whenever the optimal agent reaches the target the hybrid agent reaches it as well. In this case its expected cost would be $E[C|S]$. Wherever the optimal agent stops, the hybrid agent follows the sophisticated agent's actions. Assume that the hybrid agent has incurred some cost K in reaching a node u at which the optimal agent stops. At this point, using the decision rule for the sophisticated agent, we know that $\pi_s(u) \geq -\lambda K$. In particular, this implies that the extra expected cost for continuing is at most λK. Thus, the hybrid agent can incur a total cost of at most $(1 + \lambda)K$ along this path. Taking this as an expectation over all paths in which the optimal agent stopped prematurely, we find that the expected cost incurred by the hybrid agent is at most $(1 + \lambda)E[C|\bar{S}]$. Thus, the expected payoff of the hybrid agent is at least:

$$p(S) \cdot R - p(S) \cdot E[C|S] - (1 - p(S))(1 + \lambda)E[C|\bar{S}]$$

which equals $\pi_o - \lambda \cdot (1 - p(S)) \cdot E[C|\bar{S}]$. □

By applying Proposition 3.3 we conclude that

Corollary 3.5 $\pi_s \geq \pi_o - \lambda \cdot (1 - p(S)) \cdot E[C|\bar{S}]$.

Next, we derive two more specific bounds:

Claim 3.6 $\pi_s \geq \pi_o - \lambda \cdot p(S) \cdot R$.

Proof: As we know that optimal agent always has a non-negative payoff we have that:

$$p(S) \cdot R - p(S) \cdot E[C|S] - (1 - p(S)) \cdot E[C|\bar{S}] \geq 0$$
$$\implies (1 - p(S)) \cdot E[C|\bar{S}] \leq p(S) \cdot R - p(S) \cdot E[C|S]$$
$$\implies (1 - p(S)) \cdot E[C|\bar{S}] \leq p(S) \cdot R$$

By applying Corollary 3.5 we have that $\pi_s \geq \pi_o - \lambda \cdot (1 - p(S)) \cdot E[C|\bar{S}] \geq \pi_o - \lambda \cdot p(S) \cdot R$ □

We can get a closed form bound by using the fact that $\pi_s \geq 0$:

Proposition 3.7 $\pi_s \geq \pi_o - \frac{\lambda}{1+\lambda} \cdot R$.

Proof: If $\pi_o \leq \frac{\lambda}{1+\lambda} \cdot R$, then the claim holds simply because $\pi_s \geq 0$. Else, $\pi_o \geq \frac{\lambda}{1+\lambda} \cdot R$. Thus we have that:

$$\pi_o = p(S) \cdot R - p(S) \cdot E[C|S] - (1 - p(S)) \cdot E[C|\bar{S}] \geq \frac{\lambda}{1+\lambda} \cdot R$$

By rearranging we get that:

$$(1 - p(S)) \cdot E[C|\bar{S}] \leq p(S)(R - E[C|S]) - \frac{\lambda}{1+\lambda} \cdot R$$
$$\leq R - \frac{\lambda}{1+\lambda} \cdot R \leq \frac{1}{1+\lambda} \cdot R$$

This implies that $\pi_s \geq \pi_o - \frac{\lambda}{1+\lambda} \cdot R$ as required. □

In the supplementary material we provide a complete analysis of 3-node graphs and show that for such graphs this bound is not tight. This is done by showing that for any 3-node graph $\pi_s \geq \pi_o - \frac{2+\lambda-2\sqrt{1+\lambda}}{\lambda} \cdot R$ and noting that for any $\lambda \geq 0$ we have that $\frac{\lambda}{1+\lambda} > \frac{2+\lambda-2\sqrt{1+\lambda}}{\lambda}$. The gap between the two bounds increases as λ approaches 0, hence, in the next section we focus on the case of $0 \leq \lambda \leq 1$ and present a tighter bound for this specific and natural topology. We present some evidence that this bound is, in a sense, asymptotically tight. It is interesting to note that the bound $\pi_s \geq \pi_o - \frac{\lambda}{1+\lambda} \cdot R$ is tight for any value of $\lambda \geq 0$ in an extended model in which the cost for continuing at a certain node may depend on the edge that was taken. In other words, the costs are on the edges instead of on the nodes. We provide this proof in the supplementary material as well.

4 ANALYSIS OF THE FAN GRAPH

In this section we focus on $0 \leq \lambda \leq 1$, motivated by our goal of finding tighter bounds when λ is bounded. We consider a specific graph topology which is relatively easy to analyze: the fan. A fan graph consists of a path $(s = v_1, v_2, \ldots, v_n, t)$ such that for any $1 \leq i < n$: $p(v_i, v_{i+1}) = p_i$, $p(v_i, t) = 1 - p_i$ and $c(v_i) = c_i$. Also, $p(v_n, t) = 1$ and $c(v_n) = c_n$. A sketch of a *specific* fan is depicted in Figure 5. Fan graphs can be used to capture, for example, scenarios of project development: at every step some cost should be invested to continue the project and then with some probability it is successful.

For fans we derive an essentially asymptotically tight bound on the difference between the payoff of a sophisticated agent and an optimal agent. We present the bound and then demonstrate its tightness:

Theorem 4.1 *In every fan graph $\pi_s \geq \pi_o - \lambda \cdot (1 - \frac{1}{n})^n \cdot R$.*

Proof: Recall that by Corollary 3.5 we have that $\pi_s \geq \pi_o - \lambda \cdot (1 - p(S)) \cdot E[C|\bar{S}]$, where S is the event that the optimal agent reached the target and $E[C|\bar{S}]$ is the expected cost of the optimal agent for paths in which it stopped before the target. To bound the payoff of the sophisticated agent we bound $(1 - p(S)) \cdot E[C|\bar{S}]$. Assume without loss of generality that the optimal agent stopped traversing at v_k. In the fan there is only a single failing path $(s, v_2, \ldots, v_k)$. Thus,

$$(1 - p(S)) \cdot E[C|\bar{S}] = \left(\prod_{i=1}^{k-1}(1 - p_i)\right) \cdot \left(\sum_{i=1}^{k-1} c_i\right) \quad (1)$$
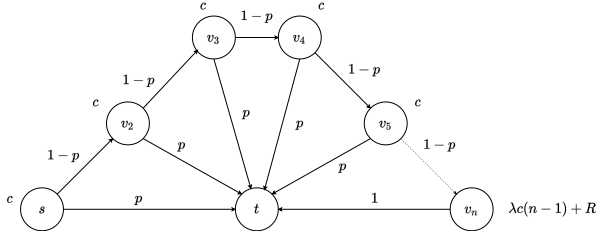
Figure 5: An instance of the Fan graph for Theorem 4.1.

We now use the fact that the optimal agent traverses the graph till v_k to get an upper bound on $(1 - p(S)) \cdot E[C|\bar{S}]$.

Lemma 4.2 *If the optimal agent reaches v_k then* $\sum_{i=1}^{k-1} c_i \leq \sum_{i=1}^{k-1} p_i \cdot R$.

Proof: Let $\pi_o(v_j, v_k)$ denote the payoff of an optimal agent traversing the graph from node v_j to v_k. We prove by a backward induction that for any $j \geq 1$, $\pi_o(v_j, v_k) \leq \sum_{i=j}^{k-1} (p_i \cdot R - c_i)$. For the base case observe that if the agent continues from v_{k-1} to v_k it is indeed the case that $\pi_o(v_{k-1}, v_k) = p_{k-1} \cdot R - c_{k-1}$. For the induction step we assume correctness for v_{j+1} and prove it for an agent traversing the graph starting from v_j. Observe that, if the agent traverses the graph from v_j to v_k then: $\pi_o(v_j, v_k) = p_j \cdot R - c_j + (1 - p_j) \cdot \pi_o(v_{j+1}, v_k)$. By the induction hypothesis we have that $\pi_o(v_{j+1}, v_k) \leq \sum_{i=j+1}^{k-1} (p_i \cdot R - c_i)$ putting this together we get that $\pi_o(v_j, v_k) \leq \sum_{i=j}^{k-1} (p_i \cdot R - c_i)$ as required. Thus, the expected payoff of the optimal agent reaching v_k is at most $\sum_{i=1}^{k-1} (p_i \cdot R - c_i)$. Since this quantity has to be non-negative, we have $\sum_{i=1}^{k-1} c_i \leq \sum_{i=1}^{k-1} p_i \cdot R$. $\square$

By applying Lemma 4.2 on Equation 1 we get that

$$(1 - p(S)) \cdot E[C|\bar{S}] \leq \left(\prod_{i=1}^{k-1}(1 - p_i)\right) \cdot \left(\sum_{i=1}^{k-1} R \cdot p_i\right)$$

In the supplementary material we show that the maximum of this function is attained when for every $1 \leq i < k$: $p_i = \frac{1}{k}$ implying

$$(1 - p(S)) \cdot E[C|\bar{S}] \leq R \cdot (k-1) \cdot \frac{1}{k} \cdot (1 - \frac{1}{k})^{k-1} = R \cdot (1 - \frac{1}{k})^k$$

thus, the maximum is attained for $k = n$. $\square$

By noticing that when n approaches infinity $(1 - \frac{1}{n})^n$ approaches $\frac{1}{e}$ we conclude that:

Corollary 4.3 *In every fan graph* $\pi_s \geq \pi_o - \frac{1}{e} \cdot \lambda \cdot R$.

Next, we show this bound is, in some sense, tight in the limit:

Theorem 4.4 *There exists a family of instances in which λ is a function of n, such that as n goes to infinity* $\lim_{n \to \infty} \pi_o - \frac{1}{e} \cdot \lambda \cdot R = \pi_s$.

Proof: We consider fan graphs in which the transition probabilities and costs on the path are identical. That is, for every $1 \leq i < n$, $p_i = p = \frac{1}{n}$ and $c_i = c = \frac{1}{n} - \frac{1}{n^2}$. We also set $R = 1$. The graph is illustrated in Figure 5.

It is easy to see that the optimal agent would traverse the graph till it reaches v_n as the expected payoff of each step by itself is strictly positive. The optimal agent would stop at v_n as the expected payoff of this last step is negative. The expected payoff of the optimal agent is

$$\pi_o = \sum_{i=1}^{n-1} (1 - p)^{i-1}(p - c) = \frac{1 - (1 - p)^{n-1}}{p}(p - c)$$

Next, we let $\lambda = \frac{\pi_o}{(1-p)^{n-1}(n-1)c}$. In Lemma **??** in the appendix, we show that it is indeed the case that $\lambda \leq 1$.

Observe that $\lim_{n \to \infty}(1 - p)^{n-1}(n-1)c = \frac{1}{e}$. Thus, we have that as n approaches infinity π_o is approaching $\frac{\lambda}{e}$. To complete the proof we show that for this value of λ, $\pi_s = 0$. To this end we show that if the agent starts traversing the graph it will go all the way to t and in this case its expected payoff will be 0. Observe that the expected payoff of a sophisticated agent that continues from node v_k till t is:

$$\pi_s(v_k) = \pi_o(v_k) - \lambda(1 - p)^{n-k}(n-1)c$$

The first term is the payoff of the sophisticated agent for getting to t before v_n which is just the same as the optimal agent and the second term is the cost that the sophisticated agent incurs for taking the last step from v_n to t. Thus, the expected payoff of a sophisticated agent that starts at s and only stops when reaching t is

$$\pi_s(v_1) = \pi_o - \lambda(1 - p)^{n-1}(n-1)c$$

which is 0 for our choice of λ.

We now prove that a sophisticated agent that starts traversing the graph will go all the way to t. We give a proof by a backward induction. That is, we assume that if the agent arrives at v_{k+1} from v_k then it will go all the way to t. For the base case observe that if the agent arrives at v_n, then the payoff for abandoning is $-\lambda c(n-1)$. On the other hand, moving to t incurs the same payoff: $R - (\lambda c(n-1) + R) = -\lambda c(n-1)$ which is the same as quitting. For ease of presentation we assume that the agent breaks ties in favor of continuing and hence the agent will choose to continue.[6] For the induction step we assume correctness for v_{k+1} and prove for an agent traversing the graph from v_k. By the induction hypothesis, since the sophisticated agent continues to traverse the graph from v_{k+1} till t then its expected payoff for continuing is $\pi_s(v_k)$. To complete the proof we show that the expected payoff is greater than or equal to the agent's sunk cost (i.e., $\pi_s(v_k) \geq -\lambda c(k-1)$):

---

[6]To avoid applying a tie-breaking rule we can add small perturbations to the costs to make sure that the agent will strictly prefer to continue.

Lemma 4.5 $\pi_s(v_k) \geq -\lambda c(k-1)$.

Proof: We need to show that:

$$\pi_o(v_k) - \lambda(1-p)^{n-k}(n-1)c \geq -\lambda c(k-1)$$

By rearranging we get that:

$$\pi_o(v_k) \geq \lambda c \left((1-p)^{n-k}(n-1) - (k-1)\right)$$

Since $\lambda c > 0$, if $((1-p)^{n-k}(n-1) - (k-1)) \leq 0$ the lemma trivially holds. Else, assume that $((1-p)^{n-k}(n-1) - (k-1)) > 0$, hence we can divide by this and get that

$$\lambda \leq \frac{\pi_o(v_k)}{c((1-p)^{n-k}(n-1) - (k-1))}$$

By substituting for our choice of λ we get that:

$$\frac{\pi_o}{(1-p)^{n-1}(n-1)c} \leq \frac{\pi_o(v_k)}{c((1-p)^{n-k}(n-1) - (k-1))}$$

By rearranging we get that:

$$\frac{1 - (1-p)^{n-1}}{(1-p)^{n-1}(n-1)} \leq \frac{1 - (1-p)^{n-k}}{(1-p)^{n-k}(n-1) - (k-1)}$$

which implies that

$$\underbrace{(1-p)^{n-k}(n-1) - (k-1) + (1-p)^{n-1}(k-1)}_{f(k)} \leq$$
$$(1-p)^{n-1}(n-1)$$

Finally, we apply Lemma 4.6 to show that the above inequality holds. This is done by proving that $f(k)$ is bounded from above by $(1-p)^{n-1}(n-1)$ for any integer $1 \leq k \leq n$. $\square$

Lemma 4.6 *The function* $f(x) = (1-p)^{n-x}(n-1) - (x-1) + (1-p)^{n-1}(x-1)$ *where* $0 < p < 1$ *is bounded above by* $(1-p)^{n-1}(n-1)$ *for each* $1 \leq x \leq n$.

Proof: In order to prove this claim we show that f is convex in $[1,n]$ and that $f(1) = f(n) = (1-p)^{n-1}(n-1)$. Therefore, for each $x \in [1,n]$, $f(x) \leq (1-p)^{n-1}(n-1)$.

Observe that indeed $f(1) = f(n) = (1-p)^{n-1}(n-1)$. In addition:

$$f'(x) = -\ln(1-p)(n-1)(1-p)^{n-x} + (1-p)^{n-1} - 1$$
$$f''(x) = \ln^2(1-p)(n-1)(1-p)^{n-x}$$

Thus, for $1 < x < n$ and $0 < p < 1$ we have that $f''(x) \geq 0$. Therefore, $f(x) \leq (1-p)^{n-1}(n-1)$ for each $x \in [1,n]$ as required. $\square$

$\square$

5 CONCLUSION AND FURTHER DIRECTIONS

Sunk cost bias is a key behavioral bias that people exhibit, and it interacts in complex ways with uncertainty about the future. We have proposed a model that allows us to study this interaction, by analyzing the loss in performance of agents that experience sunk cost bias as they perform a planning problem with stochastic transitions between states.

There are a number of further questions suggested by this work. A central open question is whether fan graphs represent the worst case for sophisticated agents with sunk cost bias. It would also be interesting to explore how the stochastic environment for sunk cost bias can be adapted to incorporate agents with other kinds of biases as well.

References

Susanne Albers and Dennis Kraft. Motivating time-inconsistent agents: A computational approach. In *Proc. 12th Workshop on Internet and Network Economics*, 2016.

Hal R Arkes and Catherine Blumer. The psychology of sunk cost. *Organizational behavior and human decision processes*, 35(1):124–140, 1985.

Elliot Aronson. Dissonance theory: Progress and problems. 1968.

Ned Augenblick. The sunk-cost fallacy in penny auctions. *The Review of Economic Studies*, 83(1):58–86, 2016.

Martin D Coleman. Sunk cost and commitment to medical treatment. *Current Psychology*, 29(2):121–134, 2010.

Richard Dawkin. The selfish gene. *Oxford university Press*, 1:976, 1976.

Jonathan D Eisenberg, H Benjamin Harvey, Donald A Moore, G Scott Gazelle, and Pari V Pandharipande. Falling prey to the sunk cost bias: a potential harm of patient radiation dose histories. *Radiology*, 263(3):626–628, 2012.

Howard Garland. Throwing good money after bad: The effect of sunk costs on the decision to escalate commitment to an ongoing project. *Journal of Applied Psychology*, 75(6):728, 1990.

Nick Gravin, Nicole Immorlica, Brendan Lucier, and Emmanouil Pountourakis. Procrastination with variable present bias. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, pages 361–361, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3936-0.

Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. *Econometrica 47,2*, pages 263–291, 1979.

Jon Kleinberg and Sigal Oren. Time-inconsistent planning: A computational problem in behavioral economics. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, EC '14, pages 547–564, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2565-3.

Jon Kleinberg, Sigal Oren, and Manish Raghavan. Planning problems for sophisticated agents with present bias. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, pages 343–360, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3936-0.

Jon Kleinberg, Sigal Oren, and Manish Raghavan. Planning with multiple biases. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, EC '17, pages 567–584, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4527-9.

R Preston McAfee, Hugo M Mialon, and Sue H Mialon. Do sunk costs matter? *Economic Inquiry*, 48(2):323–336, 2010.

Ted O'Donoghue and Matthew Rabin. Doing it now or later. *American Economic Review*, 89(1):103–124, March 1999.

Ted O'Donoghue and Matthew Rabin. Choice and procrastination. *Quarterly Journal of Economics*, 116(1):121–160, February 2001.

Gary Smith, Michael Levere, and Robert Kurtzman. Poker player behavior after big wins and big losses. *Management Science*, 55(9):1547–1555, 2009.

Barry M Staw. Knee-deep in the big muddy: A study of escalating commitment to a chosen course of action. *Organizational behavior and human performance*, 16(1):27–44, 1976.

Barry M Staw. Rationality and justification in organizational life. *Research in organizational behavior*, 2:45–80, 1980.

Pingzhong Tang, Yifeng Teng, Zihe Wang, Shenke Xiao, and Yichong Xu. Computational issues in time-inconsistent planning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3665–3671, 2017.

Richard Thaler. Toward a positive theory of consumer choice. *Journal of Economic Behavior & Organization*, 1(1):39–60, 1980.

Patrick J Weatherhead. Do savannah sparrows commit the concorde fallacy? *Behavioral Ecology and Sociobiology*, 5(4):373–381, 1979.