Need I remind you?
Monitoring with collective memory*

David A. Miller          Kareen Rozen
UCSD                   Yale

April 2009
Preliminary and Incomplete Draft

Abstract

We consider a team setting where forgetful players with limited memories have costly but socially efficient tasks to complete. Each teammate promises to complete some subset of the tasks, and strategically memorizes her own promises as well as a subset of her teammates’ promises. She can be contractually punished for an unfulfilled promise only if another player remembers it. Hence the team’s collective memory serves as a costly monitoring device.

We show that linear contracts are the optimal way to ensure that a player completes as many promises as she remembers, and characterize the optimal linear contract when players’ memories differ in size and quality. Linear contracts are optimal if players are not very forgetful. However, when players are more forgetful, the optimal equilibrium has empty promises; these are promises a player might not complete even if she remembers them. The corresponding optimal contract will “turn a blind eye” to some failures. As players become more forgetful, they make more empty promises and devote more of their memories to monitoring.

Keywords: Bounded memory, costly monitoring, team production, empty promises, collective memory, cross-cueing, transactive responsibility, optimal contracts.

*We are grateful to Itzhak Gilboa, Ben Polak, Andy Postlewaite, Larry Samuelson, and Juuso Välimäki for helpful discussions.


1 Introduction

The economic literature commonly assumes that an agent’s memory has unbounded capacity and perfect recall. By contrast, the literature in cognitive psychology has established that individual memories are imperfect, and, most importantly for models of interaction, that the collective memory of a group has very different properties than individual memory. In particular, collective memory can be generated and maintained by collaborative recall processes such as cross-cueing, by which one individual’s recall triggers a forgotten memory in another (Weldon and Bellinger (1997)). We study team production among players with imperfect memories, a question that falls into the intersection of the literatures on teams (e.g., Holmstrom (1982)), contracting with costly monitoring (e.g., Williamson (1987)), public goods (e.g., Palfrey and Rosenthal (1984)) and bounded rationality (see Rubinstein (1998)). We find that it is often optimal for players to make “empty promises” and to be “forgiven” for having done so.

We assume that agents are forgetful and have limited working memories. An agent can memorize only a limited number of tasks, and recalls each memorized task with i.i.d. probability less than one. In Section 2, we propose a model of team interaction in which the group’s collective memory serves as a costly monitoring device to enforce promise-keeping by its team members. A team of players has access to a set of socially efficient but privately costly tasks to be completed. Players make promises to each other regarding the set of tasks they will complete. Each player fills her memory strategically with some combination of her own and her teammates’ promises. A player can choose to complete only those tasks that she has not forgotten. She can be punished by the team only when someone reminds (or cross-cues) the team that she has failed to fulfill a promise. Because their memories are bounded, the players can monitor each other only at the expense of tasks they can accomplish themselves. The punishment for an unfulfilled promise takes the form of embarrassment, loss of status, or other penalty that does not enrich her teammates. The team commits to the schedule of punishments ahead of time; we call this schedule a contract.

---

1 Notable exceptions, typically in the decision-theoretic literature, include Dow (1991), Piccione and Rubinstein (1997), Hirshleifer and Welch (2001), Mullainathan (2002), Benabou and Tirole (2002) and Wilson (2003). There is also a literature on repeated games with finite automata which can be interpreted in terms of memory constraints (for example, Piccione and Rubinstein (1993), Cole and Kocherlakota (2005), and Compte and Postlewaite (2008)).

2 A seminal paper by Miller (1956) suggests that the capacity of working memory is approximately $7 \pm 2$ “chunks.” A chunk is a set of information such that associations are strong within chunks and weak across chunks—e.g., information about a task. More recently, Cowan (2000) suggests a grimmer view of $4 \pm 1$ chunks for more complex chunks. Other studies on information processing and memory include Cloitre, Cancienne, Brodsky, Dulit and Perry (1996), Tafarodi, Tam and Milne (2001), Franken, Rosso and van Honk (2003) and Tafarodi, Marshall and Milne (2003).

---
Tasks can be interpreted in two ways. In one interpretation, a task is a unitary object that is either remembered or forgotten in its entirety. Then one player’s unfulfilled promise can be discovered by another player only if he himself remembers the promise. Another, perhaps more realistic, interpretation is that each task contains detailed information, such as a decision tree, that is difficult to remember, but is necessary to complete the task properly. In this interpretation, if a player “forgets” a task, she actually forgets relevant details and is unable to complete it properly. Even if she remembers the details, by ignoring them she can “botch” the task at no cost to herself. Another player can discover that she has botched the task only if he himself remembers the relevant details. We adopt this latter interpretation, and throughout this paper use “completing a task” as shorthand for “completing a task properly.”

Our model applies to teams facing complicated tasks that must be divided up among team members, where performance is not formally contractible. Consider the following examples:

- **A medical team in a busy hospital ward.** For each patient on the ward, the team develops a treatment plan, including questions, tests, and procedures, as well as the decision framework for further action. Each doctor takes primary responsibility for carrying out the treatment plan for some subset of the patients. The doctors do not monitor each other directly, but convene as a team at the end of each day to discuss their activities.

- **A team of detectives investigating a crime.** For a detective to thoroughly interview a witness, she needs to be able to notice any details that contradict or corroborate previously collected evidence. Upon noticing such a connection, the detective can expend additional effort to follow up.

- **A legal team working on a case.** There may be many legal precedents related to a case that a team of lawyers will review while preparing. Remembering these details is important during the proceedings, for example, to argue in court in order to prevent opposing counsel from striking helpful evidence.

- **Coauthors on a research paper.** Each coauthor promises to make improvements to the paper—such as proving a conjecture, rewriting a section as agreed upon, or developing connections to related literature—which require remembering potentially complex details.

We study counting contracts, in which each player’s punishment depends on the number of her unfulfilled promises that are reported by her teammates. In Section 3, we first consider the

---

3Al-Najjar, Anderlini and Felli (2006) characterize finite contracts regarding “undescribable” events, which can be fully understood only using countably infinite statements. In this interpretation, to carry out an undescribable task properly, a player must memorize and recall an infinite statement.

4Under this interpretation, we assume that the benefit of a task is an expectation and that players cannot contract on their ex-post realized payoffs.
benchmark case of linear contracts, which treat each task independently. We fully characterize optimal symmetric linear contracts when punishments are bounded. Under a linear contract, each team member completes as many of his promised tasks as he can recall. We show that when players are very forgetful, they optimally make zero promises; but if they are not too forgetful, they optimally devote an increasing fraction of their memories to their own promises and a decreasing fraction to their teammate’s promises.

We then take up the problem of optimal non-linear contracts, in Section 4. Linear contracts are optimal in this class when the probability that players forget each promise is either very low or very high, but for intermediate forgetting probabilities it is optimal to implement a non-linear contract. In particular, optimal contracts are generally forgiving: a player who fails to fulfill a small number of promises is punished only mildly, if at all, and not enough to make her willing to fulfill all her promises if she indeed remembers them. That is, players make empty promises—promises that they do not intend to fulfill.

There are several tradeoffs in constructing an optimal contract. First, since memory is limited, memory that a player devotes to monitoring her teammate’s promises cannot be devoted to her own promises, and therefore reduces the expected number of her promises that she will remember. Second, since players are forgetful, they incur punishments with positive probability, so using punishments to induce task completion is costly. Third, although finding a large number of unfulfilled promises is an informative signal of moral hazard, it may not arise if a player completes all but a few of her promised tasks. This makes promise-keeping costly to implement, opening the door for empty promises, which also help players to remember more promises.

For arbitrary memory sizes, there is a range of parameters for which the optimal contract involves making empty promises. For small memory sizes, we have shown that every optimal contract induces cutoff strategies, in which each team member fulfills as many promised tasks as she remembers up to some cutoff (which in many cases is less than the number of tasks she promised). Moreover, both the cutoff and the total number of promises are increasing in the quality of memory; and a player optimally devotes approximately the same number of memory slots to monitoring as to empty promises. We are currently completing the proof of these results for arbitrary memory sizes.

In Section 5 we also study asymmetric two-player teams, in which players can differ in both the size of their memory and their ability to recall. Restricting attention to linear contracts, we show that the local comparative statics depend on whether the individual rationality constraints (which are relatively trivial in symmetric teams) bind; starting from symmetry, for example, increasing

---

5We primarily study two-player teams, but in symmetric environments the results apply immediately to teams of arbitrary size, since the team members can be arranged in a circle and each team member can monitor the teammate to his right.
the quality of your own memory reduces the number of promises your opponent optimally makes. Globally, holding your teammate’s characteristics fixed, increasing the quality of your own memory makes you worse off.

Our model bears interesting relations to theories in cognitive psychology and organizational behavior. Remembering a promise (i.e., remembering one’s intention to complete a task at a later point) is termed prospective memory in the theory of cognitive psychology; Dismukes and Nowinski (2007) study prospective memory lapses in the airline industry, noting that they are “particularly striking” because that industry has “erected elaborate safeguards...including written standard operating procedures, checklists, and requirements...to cross check each other’s actions.” In view of such difficulties, various theories of how to optimally store, recall, and share information have been proposed in the literature on organizational behavior; for example, consider Mohammed and Dumville (2001), Xiao, Moss, Mackenzie, Seagull and Faraj (2002) and Haseman, Nazareth and Paul (2005), which draw on the seminal work of Wegner (1987). Wegner develops the notion of transactive knowledge, the idea that while we cannot remember everything, we know who remembers what we need to know. That is, “memory is a social phenomenon, and individuals in continuing relationships often utilize each other as external memory aids to supplement their own limited and unreliable memories” (Mohammed and Dumville (2001)). Our model formalizes transactive knowledge as knowledge of strategies: knowing who is responsible for monitoring each individual. This bears a formal relationship to transactive responsibility, a concept that Xiao et al. (2002) introduce to study the division of responsibilities and cross-monitoring by trauma teams in hospitals.

We view a contract as an informal agreement that is enforced by selecting among equilibria in some unspecified continuation game. In such a context, any common knowledge event at the end of game is “contractible.” The idea that cross-cueing generates common knowledge is based on an underlying conceptual model that separates working memory, which is tightly bounded, from long-term memory, which is effectively unbounded (Baddeley (2003) reviews the relevant psychological and neurological literature). Working memory holds information that can be acted on, while long-term memory holds information that can be used to verify claims about the past. A player who has forgotten one of his promises from his working memory still holds it in his long-term memory. If another player holds his promise in her working memory, she can cross-cue him, reminding him of his promise and restoring common knowledge.

Ericsson and Kintsch (1995) note, “the primary bottleneck for retrieval from LTM [long-term memory] is the scarcity of retrieval cues that are related by association to the desired item, stored in LTM.” Here the review stage of the game provides the necessary retrieval cues. Smith (2003) shows that intending to perform a task later requires using working memory to monitor for a cue that the time or situation for performing the task has arrived.

---

6Ericsson and Kintsch (1995) note, “the primary bottleneck for retrieval from LTM [long-term memory] is the scarcity of retrieval cues that are related by association to the desired item, stored in LTM.” Here the review stage of the game provides the necessary retrieval cues. Smith (2003) shows that intending to perform a task later requires using working memory to monitor for a cue that the time or situation for performing the task has arrived.
2 The model

We first provide a loose overview of the model. Before the game starts, a contract is in place that governs the punishment each player will receive as a function of the messages sent at the end of the game. There are three stages:

1. *Promise-making.* Each player promises to complete certain tasks, and then memorizes some subset of the team’s promises. Promises are public, but memorization is private.

2. *Task-completion.* Any given promise that was stored in memory has been forgotten with some probability, independently across promises. Based on her remaining memory, each player chooses some subset of her promised tasks to complete. Task completion is private.

3. *Review.* Each player sends a public report about the tasks she completed and the promises she remembers other players made. Based on these reports, each player is punished according to the contract.

We now describe the model formally. Consider a countably infinite set of tasks $\mathcal{X}$ that can be completed by one of $n$ people, indexed by $i \in \mathcal{I} = \{1, \ldots, n\}$. Each task contains detailed information that must be stored in memory in order to complete it. Each player $i$ has a bounded memory with $M_i$ slots, each of which may be used to store a promise $(x, j) \in \mathcal{X} \times \mathcal{I}$ encoding a task $x$ and the player $j$ who promises to complete it. The same promise cannot be stored in multiple memory slots, so a player’s memory state is an element of $\mathcal{M}_i = \{m_i \subseteq (\mathcal{X} \times \mathcal{I}) \mid |m_i| \leq M_i\}$. A player reaps a benefit $b$ from each task that is completed by the team, but incurs cost $c$ for each task he completes himself. Completing any given task is efficient but a player would rather not do it; i.e., $b < c < nb$.

With a contract already in place at the outset of the game (we formalize contracts below), the players enter the promise-making stage. Each player $i$ publicly announces promises $\pi_i \subset \mathcal{X} \times \{i\}$. We assume that players enforce their equilibrium promises by applying a harsh punishment to any deviator. Given the collection of all promises, $\pi = (\pi_i)_{i \in \mathcal{I}}$, each player privately decides which of these promises to memorize. Player $i$’s memorization strategy is $\mu_i : 2^{\mathcal{X} \times \mathcal{I}} \to \Delta \mathcal{M}_i$. (Additionally, let $\mu_i((x, j); \pi)$ be the marginal probability that $\mu_i(\pi)$ assigns to $(x, j)$.) We assume that players cannot delude themselves: $\text{supp} \mu_i(\pi) \subseteq \bigcup_{j \in \mathcal{I}} \pi_j$, where $\text{supp}$ denotes the support of the distribution.

---

7 Alternatively, any number of promises can be part of a perfect Bayesian equilibrium under the following deviation response: if anyone promises a deviant set of tasks, nobody commits any promises to memory, yielding zero payoffs. Since players are indifferent to monitoring or not, this off-equilibrium play is sequentially rational.

8 Hence the memory process differs significantly from Benabou (2008), which is interested in distortions of reality.
By the task-completion stage, each promise that player \( i \) had memorized is recalled with probability \( \lambda_i \leq 1 \), independently across promises. Her resulting memory state is \( m_i \in M_i \). From among the promises that she recalls, she decides which tasks to complete, using her decision strategy \( d_i : M_i \to \Delta^X \). Her decision strategy must be measurable with respect to her memory: \( \text{supp} d_i(m_i) \subseteq m_i \cap \pi_i \). We assume that players know their strategies, as in Piccione and Rubinstein (1997), even if they have forgotten the details of some tasks. This formalizes the sentiment in Wegner (1987) that “we have all had the experience of feeling we had encoded something . . . but found it impossible to retrieve.”

At the review stage, the players observe the tasks that have been completed, and each player publicly reports the promises she recalls that other players made. Let \( A_i \subset X \times \{ i \} \) be the set of promises that player \( i \) fulfilled, and let \( \hat{m}_i \subseteq m_i \cap \pi_{-i} \) be the set of other players’ promises that player \( i \) reports. The collective memory, then, contains both the union of all completed tasks and the union of all reported promises. We assume that messages are verifiable: in line with the literature on cross-cueing (e.g., Weldon and Bellinger (1997)), a player triggers the memory of other players when he reports on the details of a task. Therefore only truthful reports are incorporated into the collective memory.

A contract, fixed at the outset of the game, is a function \( V : 2^X \times I \times 2^X \times I \to \mathbb{R}^n \), which yields a vector of punishments as a function of the collective memory. The ex-post payoff of player \( i \) is

\[
U_i = b \sum_{j \in I} |A_j| - c |A_i| + V_i \left( \bigcup_{j \in I} A_j, \bigcup_{j \in I} \hat{m}_j \right).
\]

We restrict attention to counting contracts, in which each player’s punishment depends only on the number of her unfulfilled promises that are reported by other players.

**Assumption 1** (Counting contracts). Let \( \hat{m}_{-i} \equiv (X \times \{ i \}) \cap \bigcup_{j \neq i} \hat{m}_j \), and let \( f_i \equiv |\hat{m}_{-i} \setminus A_i| \). A contract must be a counting contract of the form \( V_i(\bigcup_j A_j, \bigcup_j \hat{m}_j) = v_i(f_i) \), where \( v_i : \mathbb{I}_+ \to \mathbb{R}_- \).

Since a counting contract cannot punish a player for her report (which is verifiable), it follows that she is willing to report truthfully. The revelation principle applies in the usual way, so we focus on equilibria with truthful reporting.

**Definition 1.** A contract and a truthful perfect Bayesian equilibrium in the game it induces are (together) optimal if they yield expected payoffs that are Pareto optimal in the set of all such expected payoffs. Such a contract is also (itself) optimal.

\(^9\)Under the contracts we study, a player could never gain from performing a task she did not promise. For simplicity, we rule out the possibility of doing so.
Without (much) loss of generality, we henceforth restrict attention to the case of two players. In any \( n \)-player symmetric equilibrium in which each task is monitored by a single player, it does not matter whose memory slot is used to monitor whom. Hence it can be replicated between just two players if all responsibility for monitoring player 1 is assigned to player 2, and vice versa. In the other direction, a two-player symmetric equilibrium can be replicated among \( n \)-players by arranging team in a circle and assigning each player to monitor the teammate to her right.\(^{10}\)

3 Linear contracts

We begin by studying the benchmark case of symmetric linear contracts with a per-task punishment bound of \( v < 0 \). That is, contracts of the form \( v_i(f_i) = v f_i \), where \( v \in [v, 0] \). The main result of this section is the following theorem, which characterizes optimal symmetric linear contracts when \( M \) is even.\(^{11}\)

**Theorem 1.** Suppose \( M \) is even. Then there exist \( p^* \) and \( v^* \) (given below) such that \( v_i(f_i) = v^* f_i \) is an optimal symmetric linear contract in the symmetric environment, and in its associated optimal equilibrium each player \( i \) makes \( |\pi_i| = p^* \) promises; memorizes \( \pi_i \) with probability 1; monitors \( M - p^* \) of player \(-i\)'s promises, randomizing uniformly over memorizing each \((M - p^*)\)-element subset of \( \pi_{-i} \); completes each promise in \( \pi_i \) that she recalls; and reports what she recalls of player \(-i\)'s promises truthfully. Furthermore, if \( \lambda \geq \max\{\frac{c-b}{v}, \frac{b-c}{v}\} \), then

\[
p^* = \left\lfloor \frac{\lambda v M}{b - c + \lambda v} \right\rfloor \quad \text{and} \quad v^* = \frac{p^*(b - c)}{\lambda(M - p^*)},
\]

where \( \lfloor y \rfloor \equiv \max\{\hat{y} \in \mathbb{I} : \hat{y} \leq y\} \); otherwise \( p^* = v^* = 0 \) is optimal.

Under the optimal linear contract, each player fully utilizes all her memory slots, either for storing her own promises or for monitoring her teammate, and fulfills as many promises as she remembers. The optimal number of promises is depicted in Figure 1 as a function of the recall parameter \( \lambda \). When \( \lambda \) is very low, the players should make no promises in order to avoid virtually inevitable punishments. As \( \lambda \) rises, it reaches a threshold at which it becomes optimal to make some promises. At this threshold, monitoring is still not very effective, so each player must devote half of her memory to monitoring in order to maintain the other player’s incentives. As \( \lambda \) rises

---

10. In comparing settings with different numbers of players, the social benefit of a task should be kept constant. So if \( 2b \) is the social benefit between two players, \( 2b \)—not \( nb \)—should be the social benefit among \( n \) players.

11. There may be superior asymmetric linear contracts, but they will not differ from the optimal symmetric contract by more than a task per player. Similarly, for \( M \) odd all optimal linear contracts, symmetric or otherwise, will be close to the optimal symmetric linear contracts for \( M - 1 \) and \( M + 1 \). See footnote 12 below.
Figure 1. Optimal linear contract regimes. Here, $\lambda_{M/2} = \max\{\frac{-b}{v}, \frac{b-c}{v}\}$. All $\lambda$-ranges shown are nonempty if $-b \leq v \leq (M - 1)(b - c)$.

Further, the amount of memory devoted to monitoring decreases—and hence the optimal number of promises increases.

Proof. First we show by backward induction that every element of the strategies is sequentially rational given beliefs. First, since this is a counting contract, each player is willing to report her teammate’s promises truthfully in the review stage. Since player $i$ would be harshly punished for making the wrong promises, and cannot be punished for reporting on her teammate’s promises, her promising and memorization strategies in the promise-making stage are incentive compatible as well. Hence under consistent beliefs in the task-completion stage the incentive constraint for player $i$ to complete promise $(x, i) \in \pi_i \cap m_i$ is

$$b - c \geq \lambda \mu_{-i}((x, i); \pi)v = \lambda \min\left\{\frac{M - p}{p}, 1\right\} v. \quad (2)$$

This constraint is guaranteed by the condition $\frac{1}{2}M \leq p \leq \frac{\lambda v}{b - c + \lambda v}M$, which in turn is implied by the conditions on $\lambda$ and $p^*$ in the theorem.

Next we show that this equilibrium is optimal. First, we show that, given $p^*$ and $v$ chosen optimally, it is optimal for each player $i$ to randomize uniformly over memorizing each $(M - p^*)$-element subset of $\pi_{-i}$. Indeed, this is the unique feasible memorization strategy that satisfies each incentive constraint with equality. Any other memorization strategy must make at least one incentive constraint either violated or slack. First, if an optimal memorization strategy $\tilde{\mu}_{-i}$
caused the incentive constraint for promise \((x, i)\) to be violated, then the players could improve matters by not promising \((x, i)\) in the first place. Second, if an optimal memorization strategy \(\tilde{\mu}_i\) caused the incentive constraint for promise \((x, i)\) to be slack, then they could improve matters by marginally reducing the probability of memorizing \((x, i)\), reducing the probability of a punishment when player \(i\) forgets \((x, i)\), all without disrupting any incentive constraints.

Next we demonstrate that either \(b - c = \frac{M - p^*}{p^*}\lambda v^*\) or \(p^* = 0\). If \(0 < p^* < \frac{1}{2} M\) and the incentive constraints are satisfied, then in the promise-making stage each player can memorize all of his teammate’s tasks with probability 1 and still have at least two empty slots left over, so each player can promise an additional task for which the incentive constraint is also satisfied\(^{12}\). Hence in any optimal equilibrium in which \(p^* > 0\), we must have \(p^* \geq \frac{1}{2} M\). Therefore, assuming \(p^* > 0\), we can simplify each incentive constraint to \(b - c \geq \frac{M - p^*}{p^*}\lambda v^*\), or, equivalently, \(p^* \leq \frac{\lambda v^*}{b - c + \lambda v} M\). However, if this constraint is slack, then it would improve matters to marginally increase \(v^*\), reducing the severity of punishments (which occur with positive probability) without disrupting any incentive constraints. Hence either \(b - c = \frac{M - p^*}{p^*}\lambda v^*\) or \(p^* = 0\).

Now we consider the problem of choosing \(p^*\) and \(v^*\) optimally. Clearly, if \(p^* = 0\) then it is optimal to set \(v^* = 0\), attaining zero utility for both players. So suppose that \(p^* > 0\); then an optimal contract solves

\[
\max_{p \in I, v \in [0, V]} 2p \left( \lambda (2b - c) + (1 - \lambda)(b - c) \frac{M - p}{p - v} \right)
\]

s.t. \(\frac{1}{2} M \leq p \leq \frac{\lambda v}{b - c + \lambda v} M\). \hfill (3)

Since the incentive constraints bind, it suffices to solve

\[
\max_{p \in I} 2p \left( \lambda (2b - c) + (1 - \lambda)(b - c) \right)
\]

s.t. \(\frac{1}{2} M \leq p \leq \frac{\lambda v}{b - c + \lambda v} M\). \hfill (4)

Clearly \(\lambda \geq \frac{b - c}{V}\) is a necessary condition for this problem to have a solution. Since the objective and the constraints are linear in \(p\), it is easy to see that for \(\lambda \geq \max \left\{ \frac{b - c}{V}, \frac{c - b}{b} \right\}\) it is optimal to maximize \(p\) subject to the constraints; i.e., set \(p^* = \left\lfloor \frac{\lambda v}{b - c + \lambda v} M \right\rfloor\) and \(v^* = \frac{p^*(b - c)}{\lambda (M - p^*)}\). In contrast, for \(\lambda < \max \left\{ \frac{b - c}{V}, \frac{c - b}{b} \right\}\) the players cannot earn positive utility from this problem (if it has a solution), so it is optimal to set \(p^* = v^* = 0\).

Because the optimal linear contract treats each task separately and symmetrically, a player

\(^{12}\)Here we use the assumption that \(M\) is even. If \(M\) were odd, an optimal symmetric contract might leave the one leftover slot empty, but there would be a superior asymmetric contract in which one player uses the leftover slot to make an extra promise and the other player uses it for monitoring.
is willing to complete every task she remembers so long as she is willing to complete any single task. Note that if punishment per task were unbounded ($\nu = -\infty$) it would be possible to punish severely enough to optimally devote only one slot to monitoring and implement the maximal number of promises ($M - 1$). In the following section, in which we consider nonlinear counting contracts, we show that even if punishment can be unboundedly severe, it will not always be optimal to implement the maximal number of promises, or even to complete all promises that are recalled.

4 General counting contracts

The linearity assumption made in the previous section simplified the analysis, since a linear contract treats each task separately. However, under a linear contract there is a significant likelihood that the players will not recall all of their promises, which means they face a significant likelihood of being punished. Intuitively, a linear contract might be improved on by “forgiving” a player who completes all but the last few of her promised tasks. Of course, she will not fulfill any promises for which she will be forgiven, so some her promises will be “empty.” The drawback of such a contract is that, in the unlikely event in which she recalls all of her promises, she will not fulfill all of them. The benefit is that in the very likely event that she recalls less than all of her promises, she will not be punished too severely.

In this section we analyze non-separable contracts, in which a player’s punishment can depend in an arbitrary way on the number of her unfulfilled promises that are recalled by her teammate. The main tradeoff in designing optimal non-separable contracts is between using information efficiently and ensuring that a player recalls sufficiently many promises. To provide incentives for a player to complete any given number of recalled tasks, it is most cost-effective to use the most informative signal for punishment. Mirrlees (1974, 1999) proposed this basic intuition, but our model raises the complication that a player may be able to move the support of the monitoring distribution by fulfilling enough promises. If a player recalls a small number of promises, then being punished only for the worst outcome (the maximal number of unfulfilled promises are discovered) provides the most efficient incentives. However, if a player happens to recall a large number of promises, she may have incentive to fulfill only enough of them that the worst outcome cannot arise. Thus she may leave some promises unfulfilled; these are empty promises. A memory slot devoted to an empty promise is a memory aid: it helps the player recall more promises, yielding a first-order stochastic improvement in the number of promises she recalls. At the same time, an empty promise uses up a memory slot that could be used towards obtaining a more informative monitoring signal. The better the players’ memories, the more slots they devote to “earnest promises” and the fewer slots they need devote to monitoring and empty promises.

We show that the optimal symmetric contract takes a specific, simple form. Let $p$ be the number
of promises each player makes, and let $F$ be the number of memory slots she devotes to monitoring her teammate.\footnote{For now, we have shown properties 1, 2, 4, 6, and 8 only for $M$ sufficiently small; we are working to prove them for arbitrary $M$. Properties 3, 5, and 7 are proven for arbitrary $M$.} Each player performs as many tasks as she recalls up to a cutoff $p^*$;\footnote{Except in special cases in which the optimal contract has $p = F$, and thus full rank. In such cases (which arise only for a narrow range of $\lambda$) the players still use cutoff strategies, but the contract may be more complicated than described here. In particular, properties 2, 6, and 8 may not hold.} Each player is punished only if her teammate discovers the maximum number ($F$) of her unfulfilled promises;\footnote{Here we use the fact that randomizing over how many promises to fulfill cannot be beneficial. To randomize, the player must be indifferent. Then it is would for her to put probability 1 on the highest number in the support of her randomization.} If $\lambda$ is sufficiently high, then the optimal contract is linear with $p^* = p = M - 1$ and $F = 1$; Promises $(p)$ and cutoffs $(p^*)$ increase in $\lambda$, while empty promises $(p - p^*)$ and monitoring $(F)$ decrease in $\lambda$; Players make empty promises if and only if they make less than the maximum number of promises $(M - 1)$; The number of monitoring slots $(F)$ is the same as or one more than the number of empty promises; The incentive constraint for performing $p^*$ tasks rather than $p^* - 1$ tasks binds; Unless $p = M - 1$, all other incentive constraints are slack.

We begin by developing the problem of designing an optimal symmetric contract. Let $s : \{0, \ldots, p\} \to \{0, \ldots, p\}$ be a player’s strategy that maps the number of her promises that she recalls to the number of tasks she performs.\footnote{Lemma 1 below, establishes that players randomize uniformly over which $F$ promises to monitor and, when $a$ promises are recalled, which $s(a)$ promises to fulfill.} Naturally, the strategy must satisfy $s(k) \leq k$. To determine whether a strategy $s$ is incentive compatible, we need to consider the probability distribution over $f = |\hat{m}_i \setminus A_i|$ conditional on $s(k)$ for each $k = 0, \ldots, p$. Given $F$ and $p$, if a player fulfills $a$ of her promises, the probability that her teammate will find $f$ of her unfulfilled promises is given by the compound hypergeometric-binomial distribution (this distribution is studied in Johnson and Kotz (1985)):}
To interpret [Eq. 5] observe that in order to discover \( f \) unfulfilled promises of player \( i \), player \(-i\) must have drawn \( k \geq f \) promises from the \( p-a \) promises player \( i \) failed to fulfill, and \( F-k \) promises from the \( a \) promises player \( i \) fulfilled; this is described by a hypergeometric distribution. Of these \( k \) promises, player \(-i\) must then recall exactly \( f \); this is described by a binomial distribution.

Given a strategy \( s \), the probability of performing \( a \) tasks is

\[
t_s(a) = \sum_{a'\geq a} \mathbb{I}(s(a') = a) \binom{p}{a'} \lambda^{a'} (1 - \lambda)^{p-a'}.
\]

The incentive constraints for strategy \( s \) are

\[
\sum_{f=0}^{F} v(f) \left( g(f, s(k)) - g(f, \ell) \right) \geq (s(k) - \ell)(c - b) \quad \text{for all } \ell \leq k, \text{ and all } k.
\]

We call these “downward” constraints when \( \ell < s(k) \), and “upward” constraints when \( s(k) < \ell \leq k \).

The problem of optimally implementing strategy \( s \) at minimum cost is

\[
\max_v \sum_{a=0}^{p} t_s(a) \sum_{f=0}^{F} v(f) g(f, a) \quad \text{s.t. } v(f) \leq 0 \text{ for all } f, \text{ and Eq. 7} \tag{8}
\]

Let \( h_v(a) \equiv \sum_{f=0}^{F} v(f) g(f, a) \) be the expected punishment for fulfilling \( a \) promises. An optimal contract maximizes expected benefits net of punishments, subject to incentive compatibility:

\[
\max_{p,F,s,v} \sum_{a=0}^{p} \binom{p}{a} \lambda^{a} (1 - \lambda)^{p-a} \left( s(a)(2b - c) + h_v(s(a)) \right) \quad \text{s.t. } v(f) \leq 0 \text{ for all } f, \text{ and Eq. 7} \tag{9}
\]

Next we characterize some elementary properties of optimal contracts.

**Lemma 1** (In progress). Suppose \( \lambda \) is sufficiently high that it is optimal for the players to perform at least some tasks. Then for any optimal contract, the following are satisfied:

1. Memorize all your own promises: \( \mu_i(\pi_i) = 1 \);

2. Near full utilization: Either \( p + F = M \) or \( p = F = \left\lfloor \frac{1}{2} M \right\rfloor \);

3. Uniform monitoring: Each player randomizes uniformly over which \( F \) of her teammate’s promises to monitor;

4. Uniform task completion: Each player randomizes uniformly over which \( s(k) \) tasks to complete when she recalls \( k \) of her promises;
5. **Increasing strategies**: If \( k \geq \ell \), then \( s(k) \geq s(\ell) \);

6. **Jump to the maximum**: If \( s(k) > s(k - 1) \) then \( s(k) = k \).

We say that a strategy is **promise keeping** if \( s(a) = a \) for all \( a \leq p \), and has **empty promises** otherwise. Let \( p^* = \max_a s(a) \) be the largest number of promises that are ever fulfilled under strategy \( s \). We call \( s \) a **cutoff strategy** if \( s(a) = a \) for \( a \leq p^* \) and \( s(a) = p^* \) for all \( a > p^* \). The following lemma shows that promise keeping is optimally implemented by a linear contract, and that promise keeping is optimal among strategies satisfying \( s(p) = p \).

**Lemma 2** (Promise-keeping with linear contracts). *For any \( M \) and any \( p \), promise keeping is optimally implemented by a linear contract with \( v(f) = \frac{P}{\lambda F}(b - c) \), delivering expected social utility \( 2p(b - c + b\lambda) \).*

**Proof.** Proofs are given in the appendix.

This result is used to prove the next two theorems, the first of which shows that linear contracts are optimal when \( \lambda \) is sufficiently high. Intuitively, when \( \lambda \) is very high the players expect to recall most or even all of their promises. Since each player must devote at least one memory slot to monitoring, setting \( p = M - 1 \) maximizes the number of promises. At the same time, monitoring is very effective, so even with only one monitoring slot the punishment need not be too large to induce a cutoff of \( p^* = p \). Finally, with one monitoring slot, every task is treated identically, so the contract is linear.

**Theorem 2** (In progress). *There exists \( \lambda < 1 \) such that, for all \( \lambda \geq \lambda \), \( p^* = p = M - 1 \) under the optimal contract. Furthermore, \( F = 1 \), \( v(0) = 0 \), \( v(1) = (M - 1)^{b+c} \lambda^2 \), and all incentive constraints are satisfied with equality.*

Our next theorem shows that it is optimal to do nothing when \( \lambda \) is sufficiently low. Intuitively, when \( \lambda \) is very low the players expect to recall few or none of their promises. At the same time, monitoring is not very effective, so large punishments would be needed to induce the players to perform what few tasks they might recall. Rather than risk incurring these punishments, it is better not to do any tasks at all.

**Theorem 3** (In progress). *There exists \( \lambda > 0 \) such that, for all \( \lambda \leq \lambda \), \( p^* = 0 \) under the optimal contract. Furthermore, \( v(f) = 0 \) for all \( f \).*

Between these extremes, however, it may be optimal for players to make empty promises, as demonstrated in the next theorem.
Theorem 4 (Empty promises). For any $M$ there exists $\alpha(M) \in (1, 2)$ such that if $b < c < \alpha(M)b$, then there is $\lambda > \frac{c-b}{b}$ such that for all $\lambda \in (\frac{c-b}{b}, \lambda)$, the optimal contract involves empty promises.\footnote{Proof for $M$ odd complete; proof for $M$ even in progress.}

The result follows from Lemma 4 in the appendix, which shows that although promise-keeping in the range of parameter values above gives positive social utility, it is dominated by making roughly half as many promises and fulfilling at most one one of them.

The following theorem shows that empty promises can be optimal only when it is optimal to make less than the maximal number of promises. In other words, if the players make as many promises as possible, they should intend to follow through on them.

Theorem 5. In an optimal contract that implements fulfilling a positive number of promises, $p^* < p$ if and only if $p < M - 1$.

The intuition for this result is that empty promises serve as memory aids. By memorizing more promises than she plans to fulfill, a player attains a first order stochastic improvement in the number of tasks she will complete according to her plan. However, the corresponding increase in the number of promises she leaves unfulfilled will lead her to expect a more severe punishment unless the contract is forgiving: if it does not punish her when the other players find only a “small” number of her unfulfilled promises. But if she makes the maximal number of promises ($p = M - 1$), then the contract cannot be forgiving, since it must punish her when it finds one unfulfilled promise.

Our analysis enables us to prove the following theorem for $M \leq 5$, which states that the optimal contract implements cutoff strategies, that both the number of promises and the cutoff increase in $\lambda$, and that both monitoring and the number of empty promises decrease in $\lambda$. A specific example is visualized in Figure 2.

Theorem 6. Suppose $M \leq 5$. Then for any $\lambda$ the optimal contract implements cutoff strategies, with $p - p^* \leq F \leq p - p^* + 1$. Both $p$ and $p^*$ increase in $\lambda$, while both $F$ and $p - p^*$ decrease in $\lambda$.\footnote{Proof for $M$ odd complete; proof for $M$ even in progress.}
This is proven using our previous results and two additional lemmas in the appendix. The first main ingredient is Lemma 5, which shows (for arbitrary $M$) that if $F$ is not too large relative to the number of empty promises and a technical condition is satisfied, then a cutoff is optimal and the optimal social welfare from implementing the cutoff is given by

$$
2 \sum_{a=0}^{p} \binom{p}{a} \lambda^a (1 - \lambda)^{p-a} \left( (2b - c)s(a) + \frac{(c - b)g(F, s(a))}{g(F, p^*) - g(F, p^* - 1)} \right).
$$

(10)

The second main ingredient is Lemma 7, which shows that Eq. 10 satisfies single-crossing properties. This is illustrated in Figure 3, in which the optimal social welfare for a specific example is given by Eq. 10 for $p^* \in \{1, 2, 3, 4\}$.

We are working to extend these ideas to larger $M$. Specifically, we conjecture that cutoff strategies are always optimal, and that the monotonicity results extend (except for certain special cases that may arise from dividing memory equally between own promises and monitoring). Reaching these more general conclusions requires ruling out two possibilities that could violate the assumptions of Lemma 5 (in a region of optimality) but which do not arise for $M \leq 5$. These results have held in all our numerical computations. Illustrative examples are shown in Figure 4.
5 Asymmetric memories

In this section we study optimal linear contracts when players can differ in both memory capacity and recall probability. We denote the memory capacity and recall probability of player \( i \) by \( M_i \) and \( \lambda_i \in (0,1) \), respectively. In an asymmetric two-player setting, the optimal contract chooses \( v_i, p_i \), for \( i = 1, 2 \) to maximize

\[
(2b - c) \sum_{i=1,2} p_i \lambda_i + \sum_{i=1,2} (1 - \lambda_i) \lambda_{-i} v_i p_i \mu_{-i}((x,i);\pi) \tag{11}
\]

subject to Feasibility: \( v \leq v_i \leq 0 \) and \( p_i \in \{0,1,2,\ldots,M_i\} \)

\[
\text{IC}_i: \quad b - c \geq \lambda_{-i} v_i \mu_{-i}((x,i);\pi) \quad \text{if} \quad p_i > 0, \tag{12}
\]

\[
\text{IR}_i^*: \quad p_i \lambda_i (b - c) + p_{-i} \lambda_{-i} b + (1 - \lambda_i) \lambda_{-i} p_i v_i \mu_{-i}((x,i);\pi) \geq 0,
\]

where \( \text{IR}_i^* \) is the constraint that player \( i \) prefers the contract to autarky.

Due to the dimensionality of the problem (there are seven parameters: \( M_1, M_2, \lambda_1, \lambda_2, b, c \) and \( v \)), we relax the problem by ignoring integer constraints (e.g., \( p_i \in \{0,1,\ldots,M_i\} \)). As in Lemma 1, the players randomize uniformly over which promises to monitor, so \( \mu_{-i}((x,i);\pi) = \min\left\{ \frac{M_{-i} - p_{-i}}{p_i}, 1 \right\} \).
Then substituting each IC$_i$ into both IR$_i$ and the objective function yields a reduced form problem:

$$\max_{p_1, p_2} \left\{ (p_1 + p_2)(b - c) + (p_1 \lambda_1 + p_2 \lambda_2) b \right\}$$

s.t.  
Feasibility: $0 \leq p_i \leq M_i$

$\overline{IC}_i$:  $b - c \geq \lambda_{-i} \min\left\{ \frac{M_{-i} - p_{-i}}{p_i}, 1 \right\}$ if $p_i > 0$  

$\overline{IR}_i$:  $p_i (b - c) + p_{-i} \lambda_{-i} b \geq 0$.

The constraint $\overline{IC}_i$ incorporates the bound on punishments into IC$_i$. The solution to the reduced form problem is characterized by four parameters,

$$\frac{M_1}{M_2}, \quad \sigma_1 \equiv \frac{b - c}{\lambda_1 \gamma}, \quad \sigma_2 \equiv \frac{b - c}{\lambda_2 \gamma}, \quad \gamma \equiv -\frac{b}{\gamma} > 0,$$

where $\sigma_i \in (0, 1)$ captures the ratio of net benefit from completing a task to the expected punishment for player $-i$, and $\gamma > 0$ is the ratio of the task benefit to the maximal punishment.

**Theorem 7.** Suppose that $\lambda_i \geq \max\left\{ \frac{b-c}{\gamma}, \frac{c-b}{b} \right\}$ for $i = 1, 2$. Then the optimal contract is characterized by four binding constraints: the original IC$_1$ and IC$_2$, and two additional binding constraints determined by $\frac{M_1}{M_2}$, $\sigma_1$, $\sigma_2$, and $\gamma$ according to

<table>
<thead>
<tr>
<th>$\frac{M_1}{M_2}$</th>
<th>$\sigma_1 \sigma_2 + \gamma$</th>
<th>$\sigma_2 (1 + \gamma)$</th>
<th>$\sigma_1 \sigma_2 + \gamma$</th>
<th>$\sigma_2 (1 + \gamma)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0$</td>
<td>$\sigma_1 \sigma_2 + \gamma$</td>
<td>$\sigma_2 (1 + \gamma)$</td>
<td>$\sigma_1 \sigma_2 + \gamma$</td>
<td>$\sigma_2 (1 + \gamma)$</td>
</tr>
<tr>
<td>$\frac{1}{\sigma_1}$</td>
<td>$\sigma_1 \sigma_2 + \gamma$</td>
<td>$\sigma_2 (1 + \gamma)$</td>
<td>$\sigma_1 \sigma_2 + \gamma$</td>
<td>$\sigma_2 (1 + \gamma)$</td>
</tr>
<tr>
<td>$\sigma_2 \sigma_1 - \gamma$</td>
<td>$\sigma_1 \sigma_2 - \gamma$</td>
<td>$\sigma_2 (1 + \gamma)$</td>
<td>$\sigma_1 \sigma_2 + \gamma$</td>
<td>$\sigma_2 (1 + \gamma)$</td>
</tr>
</tbody>
</table>

For each case, the number of promises is given by

**IC**$_1$ and **IC**$_2$:  $p_1 = \frac{M_1 - \sigma_1 M_2}{1 - \sigma_1 \sigma_2}$ and $p_2 = \frac{M_2 - \sigma_2 M_1}{1 - \sigma_1 \sigma_2}$,  

**IR**$_1$ and **IC**$_1$:  $p_1 = \frac{M_2 \gamma}{\lambda_1 \sigma_2}$ and $p_2 = \frac{M_2}{1 + \gamma}$,  

**IR**$_1$ and **IC**$_2$:  $p_1 = \frac{\gamma M_1}{\sigma_1 \sigma_2 + \gamma}$ and $p_2 = \frac{\sigma_2 M_1}{\sigma_1 \sigma_2 + \gamma}$,  

**IR**$_2$ and **IC**$_1$:  $p_1 = \frac{\sigma_1 M_2}{\sigma_1 \sigma_2 + \gamma}$ and $p_2 = \frac{\gamma M_2}{\sigma_1 \sigma_2 + \gamma}$,  

**IR**$_2$ and **IC**$_2$:  $p_1 = \frac{M_1}{1 + \gamma}$ and $p_2 = \frac{M_1 \gamma \sigma_1}{1 + \gamma}$.

If $\lambda_i < \max\left\{ \frac{b-c}{\gamma}, \frac{c-b}{b} \right\}$ for some $i$ then the optimal contract has $p_1 = p_2 = 0$ and $v_1 = v_2 = 0$.  

18
Note that if $M_1 = M_2$ and $\lambda_1 = \lambda_2$, the optimal contract has both $IC_1$ and $IC_2$ binding, showing that the optimal symmetric linear contract we found in Section 3 is also the optimal contract (modulo an integer problem). Whenever $IC_i$ binds, maximal punishments ($u_i = v$) are delivered to player $i$ whenever an unfulfilled promise is discovered. If $IC_i$ is slack in Theorem 7, punishments are less severe because the IR$_i$ constraint would otherwise be violated.

To understand the role of the IR constraints, suppose that player $i$ has a larger memory than another. Without the IR$_i$ constraint, if the difference in memory size is sufficiently large, player $i$ should optimally make all the promises and player $-i$ should perform all of the monitoring. However, ensuring that the contract is individually rational for both players requires that the player $-i$ should still take on some responsibility for accomplishing tasks. The following corollary clarifies how the optimal contract and number of promises vary with the qualities of the players’ memories.

**Corollary 1.** Suppose $\lambda_i \geq \max\{\frac{c-b}{b}, \frac{c-b}{b}\}$ for $i = 1, 2$. Starting from symmetry ($M_1 = M_2$ and $\lambda_1 = \lambda_2$), a marginal improvement in the memory of player 2 (either $M_2$ or $\lambda_2$) increases the number of promises player 2 makes (reducing her utility) and increases the utility of player 1. In particular, the optimal number of promises player 1 is supposed to make decreases.

Relative to the symmetric setting, the player with the worse memory benefits not only from the greater number of promises her teammate optimally makes, but also from a reduction in the number of promises she will make. This is because in order to accomplish a greater number of tasks, she must increase her monitoring of the player with the better memory.

Let us develop graphical intuition for Theorem 7. Without loss of generality, suppose that $M_2 \geq M_1$. The problem is visualized in Figure 5 which depicts the promises of player 1 on the horizontal axis and those of player 2 on the vertical axis. These are bounded by the rectangle corresponding to their memory capacities. The requirement that $\lambda_i \geq \frac{c-b}{b}$ for $i = 1, 2$ guarantees that the set of non-zero IR promise pairs is nonempty. The requirement that $\lambda_i \geq \frac{b-c}{b}$ ensures that each $IC_i$ can be satisfied when player $-i$ monitors maximally with maximal punishments. In the case of Figure 5, the intersection of $IC_1$ and $IC_2$ occurs above the IR region. This implies that if $\lambda_1 = \lambda_2$, as in the figure, then the social indifference curves optimally select the promise levels $p_1 = 5$ and $p_2 = 7$. Hence the larger burden falls on the player with the larger memory. However, whether the intersection of $IC_1$ and $IC_2$ occurs above, below, or within the IR region depends on the parameters of the problem.

### 6 Discussion

We study a team setting where forgetful players with limited memories have costly but socially efficient tasks to complete and characterize optimal contracts when the team’s collective memory
Figure 5. Finding the optimal linear contract in the asymmetric case. The vertical axis measures promises of player 1, and the horizontal axis measures promises of player 2. Diagonal lines represent social indifference curves. The optimal contract (modulo an integer problem) implements the promise vector $\star$, which attains the highest social welfare in the intersection of the four regions bounded by IR$_1$, IR$_2$, IC$_1$, IC$_1$, M$_1$, and M$_2$. The parameters are $b = 2$, $c = 3$, $\lambda_1 = \lambda_2 = 0.7$, $\bar{v} = -2.5$, $M_1 = 9$, and $M_2 = 12$. 
serves as a costly monitoring device. We show that promise keeping is optimally implemented by linear contracts, and that linear contracts are optimal only when players are not very forgetful. Otherwise, optimal contracts induce players to make empty promises, and forgive a small number of unfulfilled promises. As players become more forgetful, they make more empty promises and devote more of their memories to monitoring.

Our framework is rich enough to lend itself to several extensions. We are particularly interested in learning whether players can be induced to truthfully reveal private information about their memory capacities. The flavor of our results may extend to interesting applications in which “recalling” a promise is interpreted as having the opportunity to implement it, which is private information. For instance, if politicians privately learn whether they can implement their campaign promises, and constituents can monitor only stochastically, then in an optimal “contract” politicians may make empty promises and constituents may forgive them for doing so.

A Appendix: Proofs

A.1 Proofs for Section 4

Proof of Lemma 2. By incentive-compatibility, to ensure that \( a \) rather than \( a - 1 \) promises are fulfilled when \( a \) are recalled, we need \( h_v(a - 1) \leq h_v(a) + b - c \). By induction, \( h_v(a) \leq h_v(p) + (p - a)(b - c) \), with \( h_v(p) = 0 \) in the best case.

Letting \( v(f) = f \frac{p}{p} (b - c) \),

\[
h_v(a) = \sum_{f=0}^{F} v(f)g(f, a) = \frac{p}{b} (b - c) \sum_{f=0}^{F} fg(f, a) = (p - a)(b - c)
\]

because the expectation of the compound hypergeometric-binomial is \( (p - a)\lambda F \frac{p}{b} \). Moreover, this contract gives expected social utility

\[
2 \sum_{a=0}^{p} \binom{p}{a} \lambda^a (1 - \lambda)^{p-a} [(2b - c)a + (p - a)(b - c)]
\]

\[
= 2p(b - c) \sum_{a=0}^{p} \binom{p}{a} \lambda^a (1 - \lambda)^{p-a} + 2b \sum_{a=0}^{p} a \binom{p}{a} \lambda^a (1 - \lambda)^{p-a}
\]

\[
= 2p(b - c + \lambda b).
\]

This is positive if \( \lambda > \frac{c - b}{b} \) and largest for \( p = M - 1 \).

Lemma 3 (Only deserved punishments). In any optimal contract, \( v(0) = 0 \).
Proof. In an optimal contract, the upward incentive constraints in Eq. 7 can be dropped as discussed earlier. Because \( g(0, a) \) is decreasing in \( a \), the downward incentive constraints can only be relaxed by imposing \( v(0) = 0 \).

Lemma 4 (Scraping by). Let \( M \geq 3 \) and suppose \( M \) is odd for simplicity. There exists \( \alpha(M) \in (1, 2) \) such that if \( 0 < b < c < \alpha(M)b \), empty promises are optimal for a nonempty open interval of \( \lambda \)'s. In particular, there exists \( \bar{\lambda} > \frac{c-b}{b} \) such that for all \( \lambda \in (\frac{c-b}{b}, \bar{\lambda}) \), completing as many promises as one remembers (for any positive number of promises smaller than \( \frac{M+1}{2} \)) is feasible and gives positive social utility, but is dominated by making \( \frac{M+1}{2} \) promises and completing only one promise whenever at least one is remembered.

Proof. Let \( p = \frac{M+1}{2} \), and \( F = \frac{M-1}{2} \). Consider implementing the strategy where exactly one task is accomplished whenever at least one is remembered. Set \( v(0) = v(1) = \cdots = v(F-1) = 0 \). This implies \( h(a) = 0 \) for all \( a > 1 \).

For doing just one task to be incentive compatible, it must be that \( h(1) - h(0) \geq c - b \) and \( h(a) - h(1) \leq (c - b)(a - 1) \) for all \( a \in \{2, 3, \ldots, p\} \). For the latter condition, it suffices that \( h(1) \geq b - c \). For the latter condition, observe that \( h(1) = v(F)g(F, 1) \) and \( h(0) = h(1)\frac{g(F, 0)}{g(F, 1)} \). Since

\[
\frac{g(F, 0)}{g(F, 1)} = \frac{(\frac{p}{F})}{(\frac{p-1}{F})} = \frac{p}{p - F},
\]

\( h(0) = \frac{p}{p-F} h(1) \). Therefore, IC requires \( h(1) \leq \frac{p}{p-F}(b - c) \). Let us set \( h(1) = \frac{2}{M-1}(b - c) \) and \( h(0) = \frac{M+1}{M-1}(b - c) \).

Therefore this contract is feasible and incentive compatible, and has expected social utility

\[
2\left[(1 - (1 - \lambda)\frac{M+1}{M-1})(\frac{2}{M-1} b - c + (1 - \lambda)\frac{M+1}{M-1}(b - c)\frac{M+1}{M-1})\right].
\]

After some algebra, this expression is larger than \( 2(M-1)(b - c + b\lambda) \) (the expected social utility from the optimal contract implementing \( M - 1 \) promises and fulfilling all those remembered) if

\[
\frac{c(M^2 - 3M) - b(M^2 - 4M + 1)}{b(M - 1)} > (1 - \lambda)\frac{M+1}{M-1} + (M - 1)\lambda.
\]

Define \( \phi : [0, 1] \rightarrow \mathbb{R} \) by \( \phi(\lambda) = (1 - \lambda)\frac{M+1}{M-1} + (M - 1)\lambda \), and note that \( \phi \) is strictly increasing. Let

\[
\bar{\lambda} = \phi^{-1}\left(\frac{c(M^2 - 3M) - b(M^2 - 4M + 1)}{b(M - 1)}\right).
\]
To show that (16) holds for \( \lambda \in (\frac{c-b}{b}, \bar{\lambda}) \), it suffices to show that \( \frac{c-b}{b} < \bar{\lambda} \), or that
\[
\frac{c(M^2 - 3M) - b(M^2 - 4M + 1)}{b(M - 1)} > \phi\left( \frac{c-b}{b} \right).
\]
After some algebra, this holds if
\[
(2 - \frac{c}{b})^\frac{M+1}{2} < \frac{2M}{M-1} - \frac{cM + 1}{bM - 1}.
\]
Define \( \hat{\phi} : [1,2] \to \mathbb{R} \) by
\[
\hat{\phi}(x) = \frac{2M}{M-1} - \frac{M+1}{M-1} - (2-x)^\frac{M+1}{2}.
\]
It can be seen that \( \hat{\phi} \) is concave, first increasing and eventually negative, with a unique \( \alpha(M) \in (1,2) \) such that \( \hat{\phi}(\alpha(M)) = 0 \). Hence the bound \( 0 < b < c < b\alpha(M) \). Moreover, \( \lim_{M \to \infty} \alpha(M) = 1 \) (details...).

Consequently, this contract dominates the linear one for any \( p \leq M-1 \); hence there are empty promises in this range.

**Lemma 5.** Suppose that \( p^* \) satisfies \( p - (p^* - 1) \geq F \) and that
\[
\sum_{a=0}^{p} t_s(a) \left( g(f,a) - g(F,a) \frac{g(f,p^*) - g(f,p^* - 1)}{g(F,p^*) - g(F,p^* - 1)} \right) \geq 0 \text{ for all } f = 1, \ldots, F - 1. \tag{17}
\]
Then the contract is suboptimal if it does not involve cutoff strategies. Moreover, the best-case punishments for implementing a cutoff strategy \( p^* \) are given by
\[
\frac{c-b}{g(F,p^*) - g(F,p^* - 1)} \left( \sum_{a=0}^{p-1} \binom{p}{a} \lambda^a(1 - \lambda)^{p-a} g(F,a) + g(F,p^*) \sum_{a=p}^{p} \binom{p}{a} \lambda^a(1 - \lambda)^{p-a} \right), \tag{18}
\]
derived by setting \( v(f) = 0 \) for \( f < F \) and \( v(F) \) high enough to make \( p^* \) indifferent to \( p^* - 1 \).

**Proof.** If a contract is optimal, we can ignore the upward incentive constraints (if any bind, then it would be optimal to do that number of tasks). Suppose that \( s \) is optimal given \( p, F \) and is not a cutoff strategy. Fixing \( s \), finding the optimal punishments is a linear programming problem. By duality theory, we know that if the primal problem is \( \max u^T y \) s.t. \( A^T y \leq w \) and \( y \geq 0 \), then the dual problem problem is \( \min w^T x \) s.t. \( Ax \geq u \) and \( x \geq 0 \); the optimal solution to one problem corresponds to the Lagrange multipliers of the other, and if feasible solutions to the dual and primal achieve the same objective value then these are optimal for their respective problems.

The relaxed problem (dropping upward incentive constraints), written in the form of the primal
problem, is given by

$$\max \sum_{f=0}^{F} (-v(f)) \sum_{a=0}^{p} -g(f, a)t_s(a) \text{ subject to}$$
$$\sum_{f=0}^{F} (-v(f))(g(f, a) - g(f, k)) \leq -(a - k)(c - b) \text{ for all } a \text{ s.t. } t_s(a) > 0 \text{ and all } k < a$$

and $$-v(f) \geq 0 \text{ for all } f = 0, 1, \ldots, F$$

The dual of this problem is then

$$\min \sum_{\{(k,a) | t_s(a) > 0, k < a\}} -(a - k)(c - b)x_{ka} \text{ subject to}$$
$$\sum_{\{(k,a) | t_s(a) > 0, k < a\}} x_{ka}(g(f, a) - g(f, k)) \geq -\sum_{a=0}^{p} g(f, a)t_s(a) \text{ for all } f = 0, 1, \ldots, F$$

and $$x_{ka} \geq 0 \text{ for all } f = 0, 1, \ldots, F$$

Let $$v(f) = 0 \text{ for all } f = 0, 1, \ldots, F - 1$$, and set $$v(F) = \frac{c - b}{g(F, p^*) - g(F, p^* - 1)} \sum_{a=0}^{p} g(F, a)t_s(a)$$, which makes the IC constraint bind in comparing $$p^*$$ and $$p^* - 1$$ tasks. We know the denominator is strictly negative by the assumption that $$p - (p^* - 1) \geq F$$ and the fact that $$g(F, a) \leq g(F, a - 1)$$ for all $$a = 1, 2, \ldots, p$$. This is feasible in the primal because all downward IC constraints will be slack after the first that binds, since $$g(F, \cdot)$$ has a MLRP (or by preservation of convexity in Lemma 6). Then the value of the primal is given by

$$\frac{c - b}{g(F, p^*) - g(F, p^* - 1)} \sum_{a=0}^{p} g(F, a)t_s(a).$$

Let $$x_{ka} = 0$$ for all pairs $$(k, a)$$ except for $$a = p^*$$ and $$k = p^* - 1$$, since those IC constraints in the primal are slack. Let

$$x_{p^*, p^* - 1} = -\frac{\sum_{a=0}^{p} g(F, a)t_s(a)}{g(F, p^*) - g(F, p^* - 1)},$$

corresponding to the constraint for $$F$$ binding, since $$v(F) < 0$$. This is feasible in the dual by the assumption in (17). Then the value of the dual is the same as that in the primal, which means that the optimal punishment involves $$v(f) = 0 \text{ for all } f = 0, 1, \ldots, F - 1$$ and $$v(F) = \frac{c - b}{g(F, p^*) - g(F, p^* - 1)}.$$.

However, because all downward IC constraints are satisfied, if $$s$$ is not a cutoff strategy then at least one of the upward IC constraints that were dropped is violated, a contradiction to being an optimal strategy given $$p$$ and $$F$$. \hfill \Box

**Proof of Theorem 5** By a similar argument as in Lemma 2 whenever $$0 < p^* = p$$ the contract
should be a linear one, optimally with $p = M - 1$. Suppose that $p^* < p = M - 1$. To implement $p = M - 1$, it must be that $F = 1$. Then the hypotheses of [Lemma 5] are satisfied, and the contract optimally implementing this has $v(0) = 0$ and $v(1)$ set to make doing $p^*$ tasks indifferent to doing $p^* - 1$ tasks: that is, $v(F) = \frac{c-b}{g(F,p^*)-g(F,p^*-1)}$. Then the expected punishment when $a$ tasks are done is given by

\[(c-b)\frac{g(F,a)}{g(F,p^*)-g(F,p^*-1)} = (c-b)\frac{(M-1-a)}{(M-1-p^*)} - (M-1-a).
\]

Consequently, expected punishment is independent of $p^*$, and decrease in $a$. Because benefits are also increasing in $a$, the contract is dominated by complete promise-keeping. Promise-keeping, in turn, is dominated by not keeping any promises if $\lambda < \frac{c-b}{b}$.

To introduce the next lemma, let $q_x \in \Delta(\{0,1,\ldots,F\})$ be a probability distribution over $\{0,1,\ldots,F\}$, parametrized by $x$. Suppose $v : \{0,1,\ldots,F\} \rightarrow \mathbb{R}_-$ is decreasing ($v(f) < v(f-1)$ for all $f \geq 1$). We say $v$ gets progressively worse (better) in $f$ if $v(f-1) - v(f) > (v(f-2) - v(f-1))$ for all $f \geq 2$. If $x$ takes integer values, we say that expected punishment $h(x) = \sum_{f=0}^F v(f)q_x(f)$ gets progressively worse (better) in $x$ if $h(x) - h(x-1) < (h(x-1) - h(x-2))$ for all $x \geq 2$; if $x$ is a continuous parameter, we say expected punishment gets progressively worse (better) in $x$ if it is concave (convex) in $x$.

**Lemma 6.** Let $q_x \in \Delta(\{0,1,\ldots,F\})$ and let $q_x$ be first-order stochastically improving in the parameter $x$. Suppose that each lower-truncated expectation $\sum_{f=n}^F fq_x(f)$ has increasing differences (i.e., $\sum_{f=n}^F f(q_x(f) - 2q_x(f) + q_{x-2}(f)) \geq 0$ for all $x \geq n$) or is concave in $x$ for all $n$ (if $x$ is continuous). If $\sum_{f=0}^F v(f)q_x(f)$ is linear in $x$, then expected punishment $\sum_{f=0}^F v(f)q_x(f)$ gets progressively worse (better) in $x$ if the punishment schedule $v$ gets progressively worse (better) in $f$.

**Proof.** We prove this for $v$ getting progressively worse; the other case is similar. For any $\varepsilon \geq 0$, let $\varepsilon(n)$ be an $F + 1$-dimensional vector of the form

\[\varepsilon(n) = (0,\ldots,0,\varepsilon,2\varepsilon,\ldots,(F+1-n)\varepsilon), \quad \text{with } n \text{ zeros.}\]

Then there exists a sequence $\{\varepsilon_n\}_{n=0}^F$ such that $v = -\sum_{n=0}^F \varepsilon_n(n)$. Then it suffices to show the result for $-\varepsilon_n$ for each $n$. For $n = 0$, linearity suffices. For $n > 0$, $-\sum_{f=n}^F \varepsilon q_x(f)$ gets progressively worse in $x$ precisely if the desired condition on the lower truncated distribution holds.

**Lemma 7.** For any $M$ and $p$,

A more general mathematical result along these lines appears in Fishburn (1982).
1. The value of Eq. 10 is strictly increasing and concave in \( \lambda \).

2. If \( p < M - 1 \) and \( p_1^* < p_2^* \leq p - F + 1 \), the value of Eq. 10 for \( p_2^* \) strictly single crosses the value of Eq. 10 for \( p_1^* \) from below, as a function of \( \lambda \).

Proof. Concavity can be shown to follow from Lemma 6. The benefit of each task is linear in \( a \), increasing in \( p^* \) and independent of \( \lambda \), which is a parameter of first-order stochastic dominance for the binomial distribution. Similarly, we need only check that the expected punishment for completing \( a \) tasks,

\[
\frac{(c - b)g(F, a)}{g(F, p^*) - g(F, p^* - 1)},
\]

has increasing differences in \( a \) and \( p^* \), since \( \lambda \) cancels out of the above. The first difference is

\[
\frac{(c - b)(g(F, a) - g(F, a - 1))}{(g(F, p^*) - g(F, p^* - 1)) - (g(F, p^*) - g(F, p^* - 1))}
\]

Since \( g(F, a) \) is decreasing in \( a \), it suffices to show that

\[
\left( \frac{p - p^*}{F} \right) - \left( \frac{p - p^* + 1}{F} \right) > \left( \frac{p - p^* + 1}{F} \right) - \left( \frac{p - p^* + 2}{F} \right).
\]

By definition of the binomial coefficient,

\[
\left( \frac{p - p^*}{F} \right) = \left( \frac{p - p^* + 1}{F} \right) \cdot \frac{p - p^* + 1 - F}{p - p^* + 1}
\]

and similarly,

\[
\left( \frac{p - p^* + 2}{F} \right) = \left( \frac{p - p^* + 1}{F} \right) \cdot \frac{p - p^* + 2 - F}{p - p^* + 2 - F}.
\]

Making these substitutions in (19) we obtain the condition \( 1 < \frac{p - p^* + 1}{p - p^* + 2 - F} \), which is evidently true whenever \( p - p^* + 1 \geq F \) and \( F > 1 \) (i.e., \( p < M - 1 \)).

Proof of Theorem 6. The first nontrivial case is \( M = 3 \), in which, by Lemma 1, the only possible promise levels are \( p = 1 \) (with \( F = 1 \)) and \( p = 2 \) (with \( F = 1 \)). In both cases Theorem 5 implies the contract must be promise-keeping.

For the case \( M = 4 \), by Lemma 1 the only possible promise levels are \( p = 2 \) (with \( F = 2 \)) and \( p = 3 \) (with \( F = 1 \)). Theorem 5 implies that the last case again reduces to promise-keeping with linear contracts, and that \( p = 2 \) (with \( F = 2 \)) is suboptimal unless it is a cutoff strategy with \( p^* = 1 \). In this case the assumptions of Lemma 5 are satisfied.

Finally, for the case \( M = 5 \), by Lemma 1 the only possible promise levels are \( p = 2 \) (with \( F = 2 \)), \( p = 3 \) (with \( F = 2 \)), and \( p = 4 \) (with \( F = 1 \)). The last case again reduces to promise-keeping with
linear contracts by Theorem 5. In light of Lemma 1, strategies must be increasing for the contract to be optimal, and by Theorem 5, they cannot have empty promises if $p^* = p$. Then there is only a cutoff strategy remaining for $p = 2$, with $p^* = 1$ (same as for $M = 4$). Moreover, there is only one non-cutoff strategy for the case that $p = 3$ that could potentially be optimal: $s(a) = 0$ for $a < 2$, and $s(a) = 2$ for $a \geq 2$. To rule this out, observe that the assumptions in Lemma 5 are satisfied for $p = 3$ and $M = 5$, so a non-cutoff strategy cannot be optimal. The cutoff strategies $(p, p^*)$ remaining are given by $(x, 0), (2, 1), (3, 1), (3, 2), (4, 4)$ are potentially optimal. We know by the single crossing result for fixed $p = 3$ that $(3, 2)$ single crosses $(3, 1)$ from below. By Lemma 7 the value functions for each $p^*$ are concave in $\lambda$, so that once the value function for $p = 4$ is optimal it remains so.

A.2 Proofs for Section 5

Proof of Theorem 7. Define $\sigma_1 \equiv \frac{b-c}{\lambda_1 \underline{v}}, \sigma_2 \equiv \frac{b-c}{\lambda_2 \underline{v}},$ and $\gamma \equiv \frac{-b}{\underline{v}}$. Using this notation,

$$\text{IC}_1 \iff p_2 \leq M_2 - p_1 \sigma_2 \text{ whenever } p_1 \geq M_2 - p_2,$$

$$\text{IC}_2 \iff p_2 \leq \frac{1}{\sigma_1} (M_1 - p_1) \text{ whenever } p_2 \geq M_1 - p_1.$$

Under the assumption that $\lambda_i \geq \frac{b-c}{\underline{v}}$ we know $\sigma_i \in (0,1)$ and $\text{IC}_i$ is satisfied in the region $p_i \leq M_i - p_i$ for $i = 1, 2$. Next, observe that

$$\text{IR}_1 \iff p_2 \geq \frac{\sigma_2}{\gamma} p_1,$$

$$\text{IR}_2 \iff p_2 \leq \frac{\gamma}{\sigma_1} p_1.$$

For the individually rational region to be nonempty, one needs $\sqrt{\lambda_1 \lambda_2} \geq \frac{c-b}{b}$, which is satisfied by the assumption $\lambda_i \geq \frac{c-b}{b}$ for $i = 1, 2$.

The intersection of $\text{IC}_1$ and $\text{IC}_1$, using the form those take in the region $\{(p_1, p_2) \mid p_2 \geq M_1 - p_1, \ p_1 \geq M_2 - p_2\}$, is given by

$$p_1 = \frac{M_1 - \sigma_1 M_2}{1 - \sigma_1 \sigma_2}, \quad p_2 = \frac{M_2 - \sigma_2 M_1}{1 - \sigma_1 \sigma_2}.$$

This intersection occurs above $\text{IR}_1$ if, plugging $p_1$ above into $\text{IR}_1$, we have

$$\frac{M_2 - \sigma_2 M_1}{1 - \sigma_1 \sigma_2} \geq \frac{\sigma_2}{\gamma} \frac{M_1 - \sigma_1 M_2}{1 - \sigma_1 \sigma_2},$$

or when $\frac{M_1}{M_2} \leq \frac{\gamma + \sigma_1 \sigma_2}{\sigma_2 (\gamma + 1)}$, and is below $\text{IR}_1$ otherwise.
Similarly, the intersection occurs below $\text{IR}_2$ if
\[
\frac{M_2 - \sigma_2 M_1}{1 - \sigma_1 \sigma_2} \leq \frac{\gamma}{\sigma_1} \frac{M_1 - \sigma_1 M_2}{1 - \sigma_1 \sigma_2},
\]
or when $\frac{M_1}{M_2} \geq \frac{(1 + \gamma) \sigma_1}{\sigma_1 \sigma_2 + \gamma}$, and is above $\text{IR}_2$ otherwise.

The slope of $\text{IC}_1$ when it binds is $-\sigma_2$ and the slope of $\text{IC}_2$ when it binds is $-\frac{1}{\sigma_1}$. The social objective takes the form
\[
(b - c)(\frac{\sigma_1 - \gamma}{\sigma_1} p_1 + \frac{\sigma_2 - \gamma}{\sigma_2} p_2)
\]
and has slope $-\frac{\sigma_2 \sigma_1 - \gamma}{\sigma_1 \sigma_2 - \gamma}$. The solution is then obtained by comparing slopes in each case.

References


