Paying With Promises

Dynamic Contracting with Ex Post Participation

Mitchell Watt
Monash University

25 November 2026

2026 Econometric Society Australasia Meeting

Motivation

Many relationships between a principal and an agent are governed by incomplete contracts.

This paper studies contracts with **ex post uncertainty** and **fixed payment terms**.

- E.g., legislated pay scales for teachers, master supply agreements, and trucking "rate cards" set payment terms before individual assignments are known.

Principals can condition assignments on the history of past allocations: **the promise of good jobs in the future can induce agents to take bad jobs today**.

- E.g., Queensland teachers who serve in rural areas receive priority for preferred locations; the U.S. Navy pairs sea tours with shore duty; taxi drivers compensated for short trips by queue priority.

This Paper

I study a principal who assigns projects to agents under **fixed payment terms**.

- Each period, up to K projects can be assigned among $N \geq K$ agents.
- Project costs are realised and publicly observed.
- A task's **net flow value** for the agent in that period may be **positive or negative**.
- The principal benefits from completed projects and can commit to future assignments.
- An agent sees her assignment and promised future treatment, then may accept, refuse, or quit.

This Paper

I study a principal who assigns projects to agents under **fixed payment terms**.

- Each period, up to K projects can be assigned among $N \geq K$ agents.
- Project costs are realised and publicly observed.
- A task's **net flow value** for the agent in that period may be **positive or negative**.
- The principal benefits from completed projects and can commit to future assignments.
- An agent sees her assignment and promised future treatment, then may accept, refuse, or quit.

Questions.

1. How should the principal use future assignments to induce agents to accept costly projects?
2. How does the relationship evolve in the long run?

Related Work

- **Payment terms and future business in practice.** Harris and Nguyen (2025); Blouin and Macchiavello (2019); Macchiavello and Morjaria (2015); Bernstein (2015); Chen et al. (2024).
- **Repeated projects and relational contracts.** Forand and Zápal (2020); Samuelson and Stacchetti (2017); Andrews and Barron (2016).
- **Assignment as an incentive.** Castro et al. (2021); Ji and Vong (2026); Eliaz et al. (2025).
- **Future allocation rights.** Bird and Frug (2019); Guo and Hörner (2018); Combe et al. (2025).
- **Recursive contracts.** Spear and Srivastava (1987); Abreu et al. (1990).

This paper: commitment to general, history-dependent assignment mechanisms; ex post participation after the realised assignment; potentially different discount factors for the principal and agents.

▸ Paper comparisons

Model

Model

A principal faces K indivisible projects every period and can assign them to $N \geq K$ ex ante identical agents.

- Each agent can complete at most one project per period, and each project can be assigned to at most one agent.
- The master contract fixes the payment $w_{k,t}$ for completing project k. The completion cost $c_{k,t}$ is publicly observed.
- The project's **value to the agent** is $v_{k,t} = w_{k,t} - c_{k,t}$, the same for every agent who could do it.
- Values are i.i.d. across projects and dates, with a continuous positive density on $[\underline{v}, \overline{v}]$, where $\underline{v} < 0 < \overline{v}$.

Timing

At date zero, before any project values are realised, the principal commits to a history-dependent assignment policy. It specifies assignments for every public history h_t and value vector $\mathbf{v}_t$.

In each period t

1. The project values $\mathbf{v}_t = (v_{1,t}, \ldots, v_{K,t})$ are realised and publicly observed.
2. The principal offers assignments according to the committed policy.
3. Each assigned agent observes the assignment and its consequences for future assignments. She then accepts, rejects, or exits.
4. Accepted projects are completed. Rejected and unassigned projects disappear. An agent who exits gets zero from then on.

Payoffs

Let $a_{ik,t} = 1$ if agent i completes project k at date t. The principal gets 1 from each completed project. The agent who completes it gets $v_{k,t}$.

- **Principal**, with discount factor δ_P:

$$\mathbb{E}\left[(1-\delta_P)\sum_{t\geq 0}\delta_P^t\sum_{i,k}a_{ik,t}\right].$$

- **Agent** i, with common agent discount factor δ_A:

$$\mathbb{E}\left[(1-\delta_A)\sum_{t\geq 0}\delta_A^t\sum_{k}v_{k,t}a_{ik,t}\right].$$

An indifferent agent accepts the principal's proposal.

Contracts

The principal commits at date zero to a **history-dependent assignment rule** within the fixed payment terms.

For every public history h_t and realised value vector $\mathbf{v}_t$, the rule specifies:

1. Which projects to assign and to whom. The rule may use lotteries.
2. How future assignments depend on who accepts and who rejects.

The history, consisting of realized project values, assignments, and acceptance decisions, is public.

Recursive Formulation

We reformulate the dynamic contract problem recursively in terms of promised utility $u = (u_1, \ldots, u_N)$.

Before the current values are drawn, the contract guarantees agent i at least u_i in expected normalised discounted utility from today onward. We start at $u = 0$.

At state u, the recursive policy specifies the following for each possible $\mathbf{v}$.

- A feasible matching $X(\mathbf{v}; u)$, with $X_{ik} = 1$ if agent i is assigned project k.
- A feasible promise vector $z(\mathbf{v}; u)$ from next period onward.

We can restrict attention to rules under which assigned agents accept on path and an agent who rejects gets no future assignments. The recursive problem imposes participation to make accepting optimal.

Feasible Promise Polytope

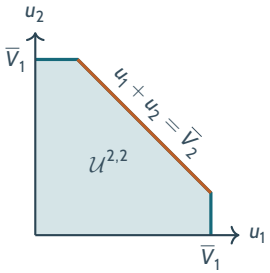
Which vectors of promised utilities can an assignment rule deliver?

Let $\overline{V}_m$ be the expected sum of the best $\min\{m, K\}$ project values, counting only positive values.

A group of m agents can receive at most that much utility per period. These bounds are also sufficient:

$$\mathcal{U}^{N,K} = \left\{ u \in \mathbb{R}_+^N : \sum_{i \in S} u_i \leq \overline{V}_{|S|} \quad \forall S \subseteq \{1, \ldots, N\} \right\}.$$

The polytope is symmetric and downward closed.



▸ Order statistics

Bellman Problem

Given u, choose rules $X(\cdot)$ and $z(\cdot)$ specifying a feasible matching and next-period promises $z \in \mathcal{U}^{N,K}$ for each possible draw. Write $Y_i = \sum_k v_k X_{ik}$ for agent i's current utility.

$$\Phi(u) = \max_{X(\cdot), z(\cdot)} \mathbb{E}_{\mathbf{v}} \left[(1-\delta_P) \sum_{i,k} X_{ik} + \delta_P \Phi(z) \right].$$

Promise keeping. Deliver each agent's promise on average over the current draw.

$$\mathbb{E}_{\mathbf{v}}[(1-\delta_A) Y_i + \delta_A z_i] \geq u_i \qquad \text{for each } i.$$

Ex post participation. After observing her assignment, an agent must weakly prefer accepting to getting zero from rejecting:

$$z_i \geq \ell(v_k) \quad \text{if } X_{ik} = 1, \qquad \ell(v) = \left[-\frac{1-\delta_A}{\delta_A} v \right]_+.$$

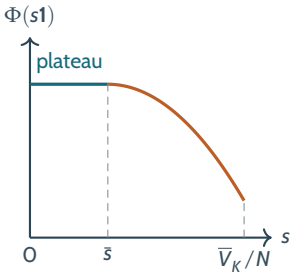
Value Function Properties

Existence and uniqueness

The Bellman operator is a δ_P-contraction and has a unique bounded fixed point Φ. An optimal deterministic Markov policy exists.

- Φ is continuous, concave, symmetric, and coordinatewise nonincreasing.
- The **plateau** is the set of promises with $\Phi(u) = \Phi(0)$. It contains strictly positive promises.
- Along any ray from zero, value is flat on the plateau and **strictly decreases** once the ray leaves it.

Equal promises $u = s\mathbf{1}$



Why flat at first? The principal wants to assign positive-value projects even at $u = 0$. Those projects deliver utility the agents were not initially promised.

▸ Single-agent value function

Results

1 Which Projects Get Assigned?

At promise state u, rank the realised projects from highest to lowest agent value:

$$v_{[1]} \geq v_{[2]} \geq \cdots \geq v_{[K]}.$$

Cutoff in rank

An optimal rule assigns all positive-value projects. After almost every draw $\mathbf{v}$, it assigns the best $m(u, \mathbf{v})$ projects, for some $m(u, \mathbf{v}) \in \{0, \ldots, K\}$.

Replacing an assigned project with a better unassigned project keeps the principal's completion payoff the same. It delivers more utility today and requires less continuation utility for acceptance.

How far down the ranking to go can depend on the promise vector and on the full draw.

▸ Conditional cutoff ▸ Single-agent contract

Optimal assignment

There is an optimal rule such that

$$u_i < u_j \quad \Longrightarrow \quad \mathbb{E}[Y_i \mid u] \leq \mathbb{E}[Y_j \mid u],$$

where Y_i is the value of agent i's project, or zero if unassigned.

- **Symmetry and concavity** imply Schur-concavity. At a fixed total promise, the principal weakly prefers a more equal distribution. Larger promises are weakly more costly at the margin, so better projects help fulfil those promises.
- **Promise keeping.** The rule must deliver each agent's promised utility. When marginal costs tie, strict priority can create too large a gap in current utility. Priority can reverse in individual draws while holding on average.

▸ Schur-concavity proof ▸ Soft priority

3 Long-Run Dynamics

Start at $u_0 = 0$ and suppose it is infeasible to assign all tasks forever with a static policy:

$$\mathbb{E}[V] < \ell(\underline{v}).$$

Let $q = \delta_A / \delta_P$. The following hold almost surely.

- **Agents more patient ($q > 1$).** Negative-value projects are assigned only finitely often.
- **Equal patience ($q = 1$).** Negative-value projects recur with positive lower frequency. Assignment approaches a history-dependent stationary policy.
- **Principal more patient ($q < 1$).** Negative-value projects recur with positive lower frequency. For an optimal rule, the agent taking the worst project repeatedly moves from the lowest to the highest priority group the next period.

▸ Static benchmark ▸ Single-agent dynamics

Deriving the Costate Equation

Let $z = u_{t+1}(\mathbf{v})$ and write $\ell_t(\mathbf{v})$ for the participation floor vector under the chosen matching. The multipliers λ_t and $\mu_t(\mathbf{v})$ price promise keeping and participation.

Lagrangian

$$\mathcal{L} = \mathbb{E}\Big[\underbrace{(1-\delta_P)\sum_{i,k} X_{ik} + \delta_P \Phi(z)}_{\text{objective}} + \underbrace{\lambda_t \cdot \big((1-\delta_A)Y + \delta_A z - u_t\big)}_{\text{promise-keeping constraint}} + \underbrace{\mu_t(\mathbf{v}) \cdot \big(z - \ell_t(\mathbf{v})\big)}_{\text{participation}}\Big].$$

Deriving the Costate Equation

Let $z = u_{t+1}(\mathbf{v})$ and write $\ell_t(\mathbf{v})$ for the participation floor vector under the chosen matching. The multipliers λ_t and $\mu_t(\mathbf{v})$ price promise keeping and participation.

Continuation FOC and envelope condition

At differentiable interior states, optimising over z and applying the envelope theorem give

$$\underbrace{-\delta_P \nabla\Phi(u_{t+1}) = \delta_A \lambda_t + \mu_t(\mathbf{v})}_{\text{continuation FOC}} \qquad \underbrace{\lambda_t = -\nabla\Phi(u_t)}_{\text{envelope}}.$$

Deriving the Costate Equation

Let $z = u_{t+1}(\mathbf{v})$ and write $\ell_t(\mathbf{v})$ for the participation floor vector under the chosen matching. The multipliers λ_t and $\mu_t(\mathbf{v})$ price promise keeping and participation.

Euler equation for the costate

Substituting $\lambda_{t+1} = -\nabla\Phi(u_{t+1})$ into the continuation FOC gives

$$\boxed{\lambda_{t+1} = q\lambda_t + \frac{\mu_t(\mathbf{v})}{\delta_P}}, \qquad q = \frac{\delta_A}{\delta_P}.$$

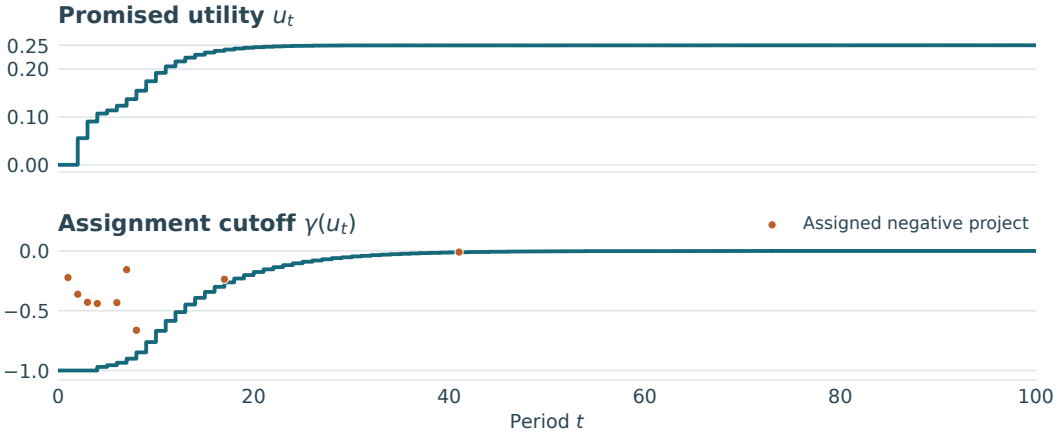
When agent i's participation constraint is slack, $\mu_{i,t}(\mathbf{v}) = 0$.

▸ Supporting costates (one agent)

Simulation

Agent More Patient ($q > 1$)

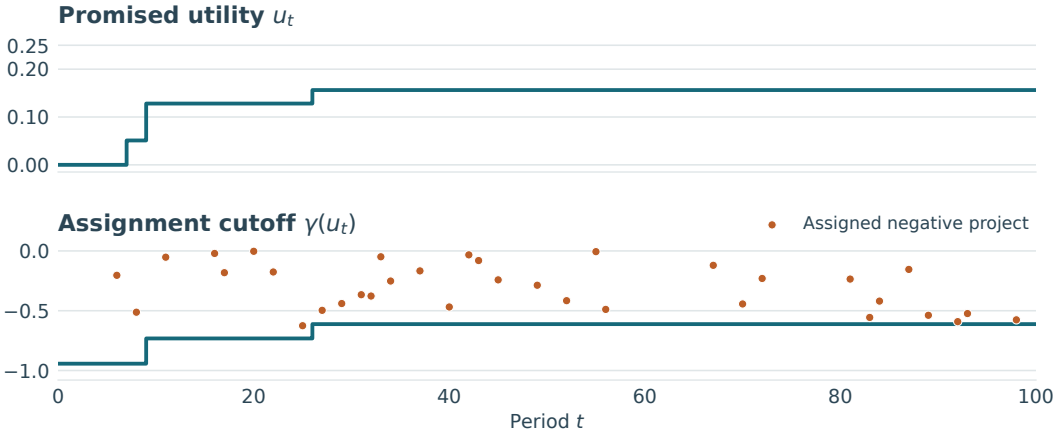
One agent, $v \sim \text{Unif}[-1, 1]$, $\delta_A = 0.8 > \delta_P = 0.7$.



Simulation

Equal Patience ($q = 1$)

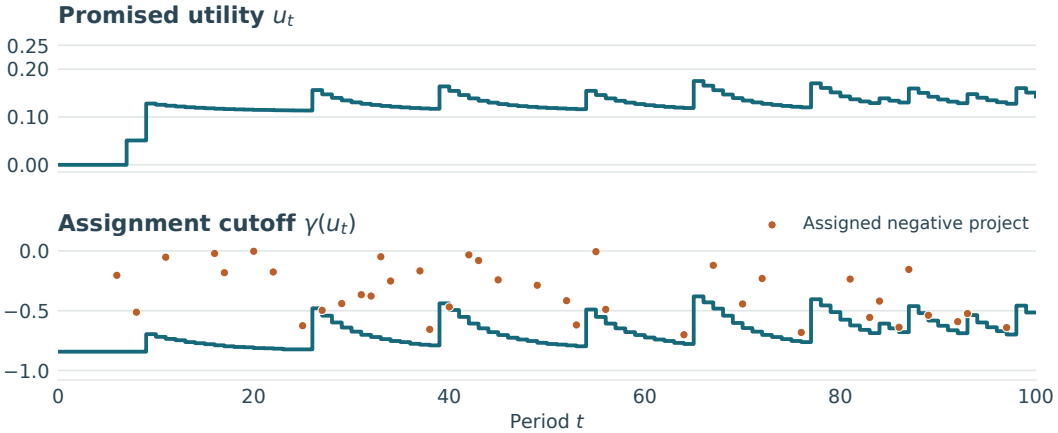
One agent, $v \sim \text{Unif}[-1, 1]$, $\delta_A = \delta_P = 0.8$.



Simulation

Principal More Patient ($q < 1$)

One agent, $v \sim \text{Unif}[-1, 1]$, $\delta_A = 0.8 < \delta_P = 0.9$.



Discussion

Skip to conclusion

The Value of Flexible Assignment

Compare K separate relationships, each tied to one project stream, with a common pool of K agents and K projects per period. All relationships start with zero promises.

Pooling creates value

With at least two project streams, pooling strictly raises the principal's payoff whenever a static rule cannot assign every project.

- An attractive project in one stream can compensate an agent for accepting a difficult project in another.
- Flexibility lets the principal direct attractive projects toward larger claims and difficult projects toward smaller claims.
- **Procurement interpretation:** pool orders across products, plants, or markets to support cooperation through future business.

The Value of Spare Capacity

Can maintaining more agents than simultaneous projects improve incentives?

- **One project per period:** extra agents create no gross value. Only the total outstanding promise matters.
- **Several projects per period:** an extra agent can strictly improve the principal's payoff. The paper establishes examples of this gain.
- **Mechanism:** spare capacity allows an agent who accepted a very difficult project to receive favourable treatment while other agents handle the remaining work.
- **Applications:** spare staffing, an additional approved supplier, or a larger pool of platform providers.

Comparative Statics: Agent Patience and Project Values

Starting from zero promises, these results apply to both one and several agents.

- **More patient agents raise the principal's payoff.** The gain is strict whenever the principal falls short of assigning all projects. Future business becomes more effective compensation for difficult work.
- **The limits are intuitive.** As $\delta_A \to 0$, only positive-value projects are assigned in the limit. As $\delta_A \to 1$, the principal's discounted assignment fraction approaches one.
- **Better project values weakly raise the principal's payoff.** If agent values improve in the sense of first-order stochastic dominance, the old assignment process remains feasible.

Comparative Statics: Principal Patience

The principal's normalised payoff P measures the discounted number of assignments per period.

- **Greater principal patience weakly lowers P.** The decline is strict when static full assignment is infeasible. More weight falls on later periods, when earlier promises must be fulfilled.
- **Total discounted assignments increase.** The unnormalised sum, $P/(1-\delta_P)$, strictly rises with principal patience.
- **The gain over the static benchmark shrinks.** A static rule delivers the same expected assignments each period; its normalised payoff is independent of principal patience.

Connection to long-run dynamics. When static full assignment is infeasible, difficult assignments recur once the principal is at least as patient as the agents.

Application

Queensland Teacher Transfers

	Year 1	Year 2	Year 3	Year 4	Year 5
Rating 7	11	11	17	20	22
Rating 6	7	7	7	10	12
Rating 5	5	5	5	7	8
Rating 4	4	4	4	6	7
Rating 3	3	3	3	4	5
Rating 2	2	2	2	2	2
Rating 1	1	1	1	1	1

Transfer points by school rating and years of service.

- Public-sector pay scales limit how much pay can vary across postings.
- Teachers earn more transfer points per year at harder-to-staff schools.
- A higher points balance gives teachers priority for future transfers.

In the model: taking a difficult assignment builds a claim on more attractive assignments later.

▸ Queensland details ▸ Other applications

Conclusion

- With payment terms fixed in advance, future projects can compensate agents for accepting costly work today.
- The principal uses attractive projects to fulfil larger promises, giving agents a form of priority that holds on average.
- When static full assignment is infeasible, costly projects keep recurring if the principal is at least as patient as the agents.

Appendix

Thank you

Less Obvious Implications

- **Patient workers need not do difficult work forever.** When static full assignment is infeasible, negative-value assignments eventually stop if agents are more patient than the principal. They recur with positive long-run frequency if the principal is at least as patient.

- **Extra workers need not improve incentives.** With one project per period, additional workers create no value. With several simultaneous projects, pooling improves on separate relationships when static full assignment is infeasible, and spare capacity can add value.

- **Priority need not be a strict queue.** An optimal rule gives workers with higher promises assignments worth at least as much on average. Tied costates can allow reversals in individual periods.

Appendix

- Single-agent model
- Static and dynamic contracts
- Single-agent dynamics
- Technical details and proofs
- Cutoffs, pooling and equalisation
- Multi-agent results
- Applications and extensions
- Related papers

Back to the talk

A Conditional Cutoff for Each Project

Fix u, a finite supporting costate vector $\lambda \geq 0$, and all other project values. Vary the value x of project k. In the one-period Lagrangian, optimise continuation promises and compare:

- $A_k(x)$, the best value among matchings that assign k. Raising x delivers more utility under the same matching and relaxes the assigned agent's participation constraint. Thus A_k is nondecreasing.
- B_k, the best value among matchings that leave k out. This value is independent of x.

Some pointwise maximiser assigns k exactly when $A_k(x) \geq B_k$. This gives a cutoff conditional on the other realised values. Promise keeping can govern selection at ties. If every costate is strictly positive, A_k strictly increases wherever assignment is feasible, so ties occur with probability zero.

Back

Static Full Assignment

A static rule uses the same matching rule every period and keeps the continuation promises fixed.

When is full assignment feasible?

A static rule can assign all K projects if and only if $\mathbb{E}[V] \geq \ell(\underline{v})$. If this holds, assign each project stream to its own agent, with a fixed promise of $\mathbb{E}[V]$.

When the condition fails, the optimal dynamic rule strictly improves on the best static rule from $u = 0$. History-dependent promises make some difficult projects worth accepting.

▸ Single-agent benchmark ◂ Back to dynamics

The Value of Pooling

Compare K separate relationships with a common pool of K agents and K project streams. All promises start at zero.

- **Pooling helps.** With at least two streams, pooling strictly raises the principal's payoff if static full assignment is infeasible. A good project in one stream can compensate for a difficult project in another.
- **Spare capacity can help.** With multiple streams, an extra agent can allow the principal to honour a large promise while others handle the remaining work.
- **One stream.** Only the total promise matters, so adding agents leaves the principal's payoff unchanged at zero initial promises.

Proof: Concavity and Symmetry Imply Schur-Concavity

Let Φ be concave and symmetric on a convex symmetric promise set. Fix a Robin Hood transfer from i to j: $\tilde{u} = u - \varepsilon e_i + \varepsilon e_j$, with $u_i > u_j$ and $0 < \varepsilon \leq (u_i - u_j)/2$.

Let σu swap coordinates i and j, and set $\alpha = \varepsilon/(u_i - u_j) \in (0, 1/2]$. Then $\tilde{u} = (1-\alpha)u + \alpha\sigma u$.

By concavity and symmetry, $\Phi(\tilde{u}) \geq (1-\alpha)\Phi(u) + \alpha\Phi(\sigma u) = \Phi(u)$.

Thus every Robin Hood transfer weakly raises the value function. Repeating the argument along any finite sequence of transfers gives $u \succeq \tilde{u} \Rightarrow \Phi(\tilde{u}) \geq \Phi(u)$.

▸ Equalisation diagram ◂ Back to results

Single-Agent Model

For now, let $N = K = 1$: **one firm**, **one worker**, and **one task** each period $t = 0, 1, 2, \ldots$.

- Each period, a task arrives with **effort cost** c_t, observed by both parties.
- The wage is **fixed** at w, so the worker's **task value** is $v_t = w - c_t$.
- Task values are i.i.d. from F on $\mathcal{V} = [\underline{v}, \bar{v}]$, where $\underline{v} < 0 < \bar{v}$ and F has a continuous strictly positive density.

The firm receives payoff 1 from a completed task; the worker receives v_t.

After observing the task and promised continuation, the worker accepts or rejects the assignment.

Timing

Single-agent model

At date zero, the firm commits to a history-dependent assignment rule before any task values are realised.

In each period

1. A task value v_t is **realised** and **publicly observed**.
2. The committed rule determines whether the firm assigns the task.
3. If assigned, the worker observes the promised continuation and **chooses whether to accept**. Otherwise, the worker receives an outside option of 0 in that period.
4. Accepted tasks are completed; rejected or unassigned tasks disappear.

Payoffs

Single-agent model

Let $x_t \in \{0, 1\}$ denote assignment and $y_t \in \{0, 1\}$ denote acceptance.

- Completion is $x_t y_t$.
- The firm receives **flow payoff** $x_t y_t$.
- The worker receives **flow payoff** $v_t x_t y_t$.
- **Discount factors** may differ: δ_P for the firm and δ_A for the worker.

The firm maximises $\mathbb{E}\left[(1-\delta_P)\sum_{t\geq 0}\delta_P^t x_t y_t\right]$, while the worker evaluates $\mathbb{E}\left[(1-\delta_A)\sum_{t\geq 0}\delta_A^t v_t x_t y_t\right]$.

Contracts

Single-agent model

A contract is a **history-dependent** assignment rule.

The contract specifies an **assignment probability** $X_t(v_t, h_t)$ for each history and task value.

The worker chooses **acceptance probability** $Y_t(v_t, h_t)$.

I impose:

- **public histories**: task values, assignments, and acceptances are observed;
- **commitment**: the firm can commit to future assignment rules;
- **tie-breaking**: an indifferent worker accepts.

The constraint of interest is ex post participation after each task realisation.

Recursive Formulation

Single-agent model

Following Abreu, Pearce and Stacchetti (1990), it is without loss to study Markov contracts with **promised continuation utility** u as a state variable.

Feasible promises lie in $\mathcal{U} = [0, V^+]$, where $V^+ = \mathbb{E}[[v]_+]$. Initial promise is zero.

At state u, the policy specifies the following for each possible v.

- assignment $X(v; u) \in \{0, 1\}$;
- continuation promise from next period onward $z(v; u) \in \mathcal{U}$.

Notes:

(a) It is without loss to restrict to **deterministic** contracts.

(b) An optimal rule can be represented with every assigned task accepted; an **ex post participation constraint** makes acceptance optimal.

(c) **Off-path**, the principal punishes the agent with the lowest feasible continuation promise (0).

Bellman Problem

Single-agent model

The firm solves:

$$\Phi(u) = \max_{\substack{X(\cdot;u):\mathcal{V}\to\{0,1\} \\ z(\cdot;u):\mathcal{V}\to\mathcal{U}}} \mathbb{E}\left[(1-\delta_P)X(v;u) + \delta_P \Phi(z(v;u))\right],$$

subject to a **promise-keeping constraint**:

$$\mathbb{E}\left[(1-\delta_A)vX(v;u) + \delta_A z(v;u)\right] \geq u \qquad \text{(PK - } \lambda(u)\text{)}$$

and an **ex post participation constraint**:

$$z(v;u) \geq \underline{u}(v)X(v;u) \quad \text{for every } v, \qquad \underline{u}(v) = \left[-\frac{1-\delta_A}{\delta_A}v\right]_+ \qquad \text{(PC - } \mu(v;u)\text{)}$$

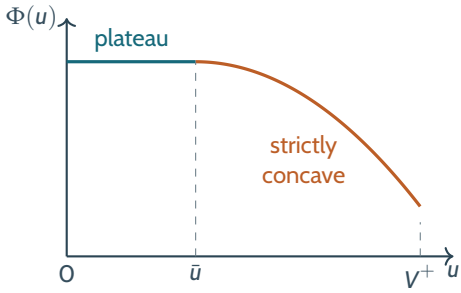
Value Function Properties

Single-agent model

Proposition 1

The Bellman problem has a unique fixed point $\Phi : \mathcal{U} \to [0, 1]$. It is nonincreasing, concave, and continuous.

- There is a **plateau** $[0, \bar{u}]$: Φ is constant there.
- Above $\bar{u}$, promises are costly: Φ is **strictly decreasing** and **strictly concave** on $(\bar{u}, V^+]$.
- Above the plateau, the marginal cost of a larger promise is positive and rises with u.



Intuition for plateau: at $u = 0$, firm wants to assign positive-value tasks ⇝ strictly positive utility (any negative value tasks assigned must be compensated by promises and contribute ≥ 0 to promised utility).

Strict concavity
Back to the main talk

Static Benchmark

Single-agent model

A **static** contract uses the same $X : \mathcal{V} \to \{0, 1\}$ after every history. Since each completed task gives the firm payoff 1, optimal static assignments are **cutoffs**: $X(v) = 1\{v \geq c\}$.

For cutoff $c < 0$, the worker's payoff from accepting the task at the cutoff is

$$H(c) = \underbrace{\delta_A \mathbb{E}[v1\{v \geq c\}]}_{\substack{\text{discounted value of} \\ \text{future assignments}}} + \underbrace{(1 - \delta_A)c}_{\substack{\text{current payoff} \\ \text{at the cutoff}}}.$$

Static cutoff c is **feasible** when $H(c) \geq 0$: future assignments compensate for taking the cutoff task.

Proposition 2: Optimal Static Contract

The **optimal static contract** is a cutoff rule $X^S(v) = \mathbf{1}\{v \geq c^S\}$, where

$$c^S = \begin{cases} \underline{v}, & \text{if } H(\underline{v}) \geq 0, \\ \text{the unique root of } H(c) = 0 \text{ in } (\underline{v}, 0), & \text{if } H(\underline{v}) < 0. \end{cases}$$

It delivers $U^S = \mathbb{E}[v1\{v \geq c^S\}]$ and $\Pi^S = \Pr(v \geq c^S)$.

When is the static contract dynamically optimal?

When $\mathbb{E}[v] \geq \frac{1-\delta_A}{\delta_A}(-\underline{v})$, the firm can assign **every task** with the fixed promise $\mathbb{E}[v] \rightsquigarrow$ optimal.

Otherwise, dynamic contracts do **strictly better**:

Proposition 4: Static Full Assignment or Dynamic Improvement

Either:

1. the optimal static contract has cutoff $c^S = \underline{v}$, so it assigns every task; or
2. the optimal static contract has cutoff $c^S > \underline{v}$, and the optimal dynamic contract strictly improves on the optimal static contract: $\Phi(0) > \Pi^S$.

- A task just below the static cutoff needs only a little extra compensation, but completing it gives the principal a whole extra task.

Henceforth, assume $\mathbb{E}[v] < \frac{1-\delta_A}{\delta_A}(-\underline{v})$, so full assignment is statically infeasible.

▸ Proof sketch

Characterising the Optimal Dynamic Contract

At state u, the firm chooses $X : \mathcal{V} \to \{0, 1\}$ and $z : \mathcal{V} \to \mathcal{U}$ to solve

$$\Phi(u) = \max_{X,z} \mathbb{E}[(1 - \delta_P)X(v) + \delta_P \Phi(z(v))],$$

subject to

$$\text{(PK - } \lambda(u)\text{)} \quad \mathbb{E}[(1 - \delta_A)vX(v) + \delta_A z(v)] \geq u, \qquad \text{(PC - } \mu(v; u)\text{)} \quad z(v) \geq X(v)\underline{u}(v) \; \forall v.$$

1. **Solve for z given X.** Fix a candidate assignment rule and solve the continuation-promise problem.
2. **Given z, show optimal X is cutoff.** After substituting optimal continuation promises, show assignment is monotone in v.
3. **Solve jointly for X and z.** Combine the two steps to characterise the optimal contract.

The Optimal Dynamic Contract

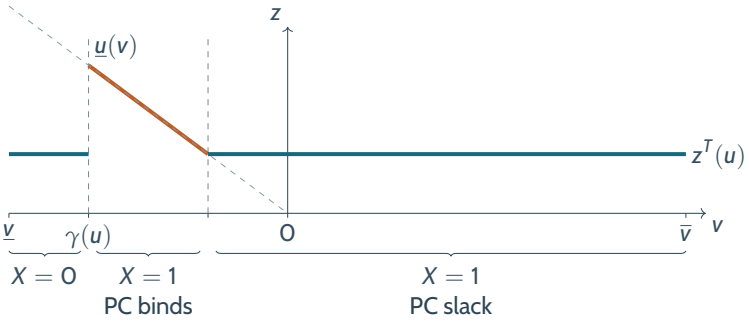
The optimal policy has the following form for some cutoff $\gamma(u)$ (equivalently, target $z^T(u)$):

Assignment:

$$X(v;u) = \mathbf{1}\{v \geq \gamma(u)\}.$$

Continuation ($\bar{u} < u < V^+$):

$$z(v;u) = \begin{cases} z^T(u), & v < \gamma(u), \\ \max\{\underline{u}(v), z^T(u)\}, & v \geq \gamma(u). \end{cases}$$



What determines the cutoff $\gamma(u)$, or equivalently the slack target $z^T(u)$?

Characterising the Slack Target

Let $\lambda \geq 0$ and $\mu(v) \geq 0$ be the promise-keeping and participation multipliers. The Lagrangian is

$$\begin{aligned}\mathcal{L} = \mathbb{E}[(1-\delta_P)X(v) + \delta_P \Phi(z(v))] \\ + \lambda\{\mathbb{E}[(1-\delta_A)vX(v) + \delta_A z(v)] - u\} \\ + \mathbb{E}[\mu(v)\{z(v) - \underline{u}(v)X(v)\}].\end{aligned}$$

FOC for $z(v)$: $\delta_A \lambda + \mu(v) \in \delta_P \partial(-\Phi)(z(v))$.

Define the **costate correspondence** $\Lambda(u) = \partial(-\Phi)(u) \cap \mathbb{R}_+$, using the subdifferential on $[0, V^+]$. The envelope condition gives $\lambda \in \Lambda(u)$: a costate is a supporting marginal cost of the promise.

Along the optimal path, $u_{t+1} = z(v_t; u_t)$ and $\lambda_t \in \Lambda(u_t)$. The FOC gives the **Euler equation**:

$$\lambda_{t+1} = \underbrace{\frac{\delta_A}{\delta_P}}_{q} \lambda_t + \frac{\mu_t}{\delta_P} \in \Lambda(u_{t+1}).$$

When participation is slack, $\mu_t = 0$ and $u_{t+1} = z^T(u_t)$, so $q\lambda_t \in \Lambda(z^T(u_t))$.

▸ Derivation

Long-Run Dynamics

Single-agent model

Relative patience $q = \delta_A / \delta_P$ determines how promises and assignments evolve through the Euler equation:

$$\lambda_{t+1} = q\lambda_t + \frac{\mu_t}{\delta_P} \in \Lambda(u_{t+1}), \qquad \mu_t \geq 0.$$

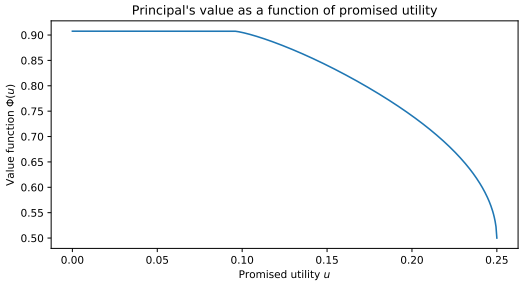
Relative patience	Promised utility	Assignments
Agent more patient $q > 1$	Approaches the feasible maximum V^+.	The cutoff approaches zero; negative-value tasks become increasingly rare.
Equal patience $q = 1$	Converges to an interior level determined by the history.	The cutoff settles at a negative level; mildly negative-value tasks remain eligible.
Principal more patient $q < 1$	Keeps fluctuating: falls when participation is slack and rises after difficult tasks.	Positive- and negative-value tasks continue to be assigned.

Back to main result

Agent More Patient: $q > 1$

Proposition 5(i) - Patient Agent

If $q > 1$, promised utility eventually rises toward its feasible maximum V^+. The assignment cutoff approaches zero, making negative-value tasks increasingly rare.

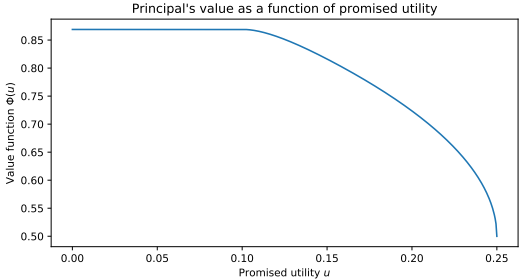


- **Exchange rate:** $q > 1$ makes future compensation relatively cheap, favouring backloading.
- In contrast, cash payments can be deferred indefinitely (Sun and Tian, 2018).
- **Here, promises are fulfilled progressively through good jobs,** because feasible promises are bounded: $u_t \leq V^+ = \mathbb{E}[[v]_+]$.

Equal Patience: $q = 1$

Proposition 5(ii) - Equal Patience

If $q = 1$, promised utility converges to an interior level determined by the history. The cutoff approaches a fixed negative threshold, so mildly negative-value tasks remain eligible for assignment.

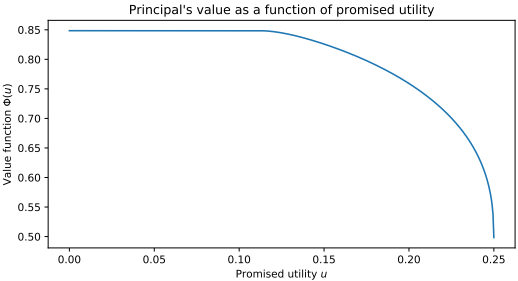


- **Equal discounting.** Both parties value the timing of compensation at the same rate. When participation is slack, $\lambda_{t+1} = \lambda_t$: the marginal cost of promises is equalised across dates.
- Difficult assignments can require additional promises, raising the costate when participation binds.
- Promises converge to an interior level determined by the history. The limiting cutoff remains negative, so mildly bad jobs continue to be assigned.

Principal More Patient: $q < 1$

Proposition 5(iii) - Patient Principal

If $q < 1$, promised utility leaves the plateau in finite time and then keeps fluctuating. It falls toward the plateau when participation is slack and rises after some difficult assignments.



- **Future compensation is relatively expensive.** The agent discounts deferred rewards more heavily, so the principal fulfils promises through current assignments and runs down future obligations.
- With participation slack, $\lambda_{t+1} = q\lambda_t < \lambda_t$, moving the target promise $z^T(u_t)$ toward the plateau.
- Difficult assignments require promises of future good jobs, raising promised utility again.
- Positive- and negative-value assignments recur as promises are fulfilled and renewed, sustaining fluctuations in the relationship.

From One Worker to Many

The only decision in the single-agent case is whether or not to assign the realised task. The multi-agent case adds the decision of who receives each assigned task.

Objects

- $N \geq 2$ workers and $K \leq N$ tasks arrive each period.
- Workers share the same discount factor δ_A.
- Task values are $v = (v_1, \ldots, v_K) \sim F^K$.
- The state is a promise vector $u = (u_1, \ldots, u_N)$.

Choices and Constraints

- Assignment is a matching matrix $X(v; u) \in \{0, 1\}^{N \times K}$.
- Each task goes to at most one worker; each worker gets at most one task.
- Promise keeping and ex post participation hold worker by worker.

Feasible Promise Polytope

Order-statistic formula

m workers can receive at most the best $\min\{m, K\}$ positive tasks per period. This bounds their total promise.

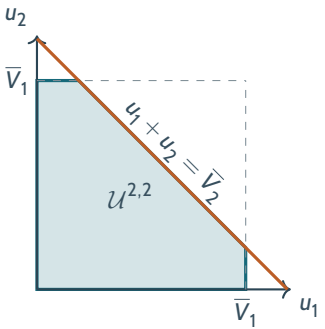
Order statistics: $v_{(1)} \le \cdots \le v_{(K)}$. For $m \le K$:

$$\overline{V}_m = \mathbb{E}\left[\sum_{r=K-m+1}^{K} [v_{(r)}]_+\right], \qquad \overline{V}_0 = 0.$$

For $m > K$, set $\overline{V}_m = \overline{V}_K$.
Feasible promises:

$$\mathcal{U}^{N,K} = \left\{ u \in \mathbb{R}_+^N : \sum_{i \in S} u_i \le \overline{V}_{|S|} \text{ for every } S \subseteq \{1, \dots, N\} \right\}.$$

A symmetric, downward-closed polytope.



Back to the model

Schur-Concavity and Equalisation

$\Phi^{N,K}$: continuous, nonincreasing, concave; a plateau. Worker **symmetry** adds a preference for equalisation.

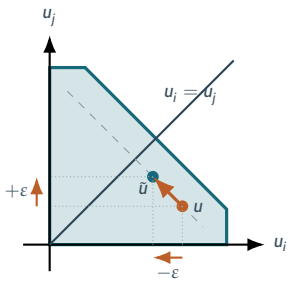
Robin Hood transfer: move promise from a worker owed more to one owed less, keeping the total fixed:

$$\tilde{u} = u - \varepsilon e_i + \varepsilon e_j, \qquad 0 < \varepsilon \leq (u_i - u_j)/2.$$

Schur-concavity: each transfer weakly raises value, $\Phi^{N,K}(\tilde{u}) \geq \Phi^{N,K}(u)$.

Majorization: $u \succeq \tilde{u}$ if a sequence of these transfers gives $\tilde{u}$. Then $\Phi^{N,K}(\tilde{u}) \geq \Phi^{N,K}(u)$.

⇝ Same total promise: more **equal** is weakly better for the principal.



▸ Schur-concavity

Optimal Assignment: Theorem and Intuition

Let $Y_i = \sum_k v_k X_{ik}$ be worker i's current task value (0 if unassigned).

Theorem 2: Assortative Allocation on Average

There is an optimal rule with

$$u_i < u_j \quad \Longrightarrow \quad \mathbb{E}[Y_i] \leq \mathbb{E}[Y_j].$$

All positive tasks assigned; negative tasks assigned least costly first.

First, we show allocation is assortative in costates. Promise keeping requires

$$\mathbb{E}[(1-\delta_A)Y_i + \delta_A z_i] \geq u_i.$$

With multiplier λ_i, its contribution to the Lagrangian is

$$\sum_i \lambda_i \big(\mathbb{E}[(1-\delta_A)Y_i + \delta_A z_i] - u_i\big).$$

Thus current utility enters as $(1-\delta_A)\mathbb{E}[\sum_i \lambda_i Y_i]$. **Higher-costate workers receive more weight**: utility delivered today saves more costly future compensation.

Why Priority is Sometimes Soft

Distinct costates: sort tasks by marginal promise cost. If $\lambda_j > \lambda_i$ and $y > x$, giving y to j rather than i raises the current-utility term by

$$(1 - \delta_A)(\lambda_j - \lambda_i)(y - x) > 0.$$

The firm's completion payoff is unchanged. Participation may require an equalising continuation transfer; its costate-weighted loss cannot exceed this gain, and continuation value weakly rises.

An optimal matching can therefore be chosen with $\lambda_i < \lambda_j \Rightarrow Y_i \leq Y_j$. Schur-concavity gives $u_i < u_j \Rightarrow \lambda_i \leq \lambda_j$.

Tied costates: only the group's current-utility total matters.

- $\lambda = 0$ supports plateau states; promises that are close to equal can also lead to tied costates $\lambda_i = \lambda_j$.
- Promise keeping still applies to each worker. When both constraints bind,

$$u_j - u_i = (1 - \delta_A)\mathbb{E}[Y_j - Y_i] + \delta_A \mathbb{E}[z_j - z_i].$$

Always giving j the better task may create too large a current gap. **Occasional reversals** keep assignments consistent with the promised gap; priority holds **on average**.

Back to results

Endogenous Priority and Equalisation

- **Priority emerges from optimal contracting.** A worker who accepts a difficult task earns a larger promise. That promise gives the worker priority for more attractive assignments, on average.
- **Equalisation has economic value.** With symmetric workers, a more even distribution of a fixed total promise weakly improves the principal's payoff. Attractive tasks help fulfil larger promises; difficult tasks create claims for workers owed less.
- **The principal maximises its own payoff.** Its objective values completed tasks. Worker symmetry and the cost of fulfilling promises generate the force toward equalisation.

Motivating Example

Allocation of School Teachers



- Teachers are allocated centrally.
- Jobs appear over time: teacher retirements, transfers, and new demand.
- Assignment values vary with locational preferences and school difficulty.
- Public-sector pay scales limit salary adjustment across postings.
- **Transfer points**: teachers accrue priority while matched to less desirable schools.

▸ More details

Queensland Teacher Transfers

	Year 1	Year 2	Year 3	Year 4	Year 5
Rating 7	11	11	17	20	22
Rating 6	7	7	7	10	12
Rating 5	5	5	5	7	8
Rating 4	4	4	4	6	7
Rating 3	3	3	3	4	5
Rating 2	2	2	2	2	2
Rating 1	1	1	1	1	1

Transfer points by school rating and years of service.

Institution.

- Teachers are allocated through a centralised transfer system.
- Vacancies arrive as teachers retire, move, or as demand changes.
- Public-sector pay scales limit salary adjustment.

Incentive instrument.

- Each year of service earns transfer points.
- Harder-to-staff schools earn more points per year.
- Vacancy priority goes to higher point balances.

Interpretation. Hardship service builds standing for future transfers.

Back to main application

Future Business as Compensation

Payment terms are fixed or slow to adjust, while fulfilment costs vary.

- **Truckload freight.** Carriers can reject loads under fixed-rate contracts; future business supports cooperation. Harris and Nguyen (2025).
- **OEM suppliers.** Master agreements precede purchase orders; scorecards influence preferred status and future business. Bernstein (2015).
- **Coffee.** World-price increases can make fixed-price forward contracts costly to fulfil; continued trade affects default. Blouin and Macchiavello (2019).
- **Kenyan roses.** Supply disruptions force exporters to prioritise orders; deliveries depend on the value of established relationships. Macchiavello and Morjaria (2015).

Back to main application

Other Assignment Currencies

Queue priority

- Uber, Lyft, and DiDi use airport or short-trip queue rules.
- A bad trip preserves or upgrades future access to better trips.

Point systems

- Amazon Flex drivers accumulate standing through completed blocks.
- Higher standing unlocks preferred stations, times, and blocks.

Rotation rules

- Foreign service and naval careers link hardship tours to later desirable postings.
- Residency programmes use point bidding for preferred rotations.

Hiring and Termination: Implications

The workforce results allow us to compare fixed pools of relationships.

- **Hiring or qualifying another supplier.** With a maintenance cost c per relationship, initial net value is $\Phi^{N,K}(0) - Nc$. An extra relationship is worthwhile when its gross benefit exceeds c.
- **Spare capacity has an incentive benefit.** With several projects, it can preserve output while earned priority is honoured. This benefit disappears once the existing pool supports full assignment every period.
- **Termination must account for earned claims.** An agent owed positive utility has a claim on future treatment. Honouring that claim sustains the incentives for earlier difficult work.

Further analysis: optimal turnover requires hiring and termination costs, and a rule for settling promises through severance, buyouts, or future business.

Extension: General Actions and Assignments

Allow general assignment and action spaces: the principal allocates opportunities and recommends observable actions; agents choose how to respond. Payment terms remain fixed.

- **Example: level of service.** Normal delivery gives a supplier utility 5; expedited delivery gives utility 2. If normal delivery remains available, expediting sacrifices $d = 3$.
- **Future allocations reward current actions.** Let d be the payoff sacrificed relative to the best available response. When deviating forfeits future access, compliance requires $z \geq (1 - \delta_A)d/\delta_A$.
- **The continuation logic carries over.** Under suitable conditions, the single-agent promise is the larger of a target and the minimum required to induce the recommended action.

Extended Conclusion

- With payment terms fixed in advance, future projects can compensate agents who accept costly assignments today.
- With one agent, the principal assigns projects above a cutoff that rises weakly with the agent's promised utility.
- Relative patience determines whether negative-value projects gradually disappear or keep recurring.
- With several agents, an optimal rule gives agents owed more assignments worth at least as much on average. Similar promises can allow reversals in individual periods.
- **Applications:** Transfer points, preferred-supplier status, and queue priority reward current service with access to future opportunities.

Solving for z given X

Fix $u \in (\bar{u}, V^+)$ and an assignment rule X. Apart from participation, v enters only through **average current utility** $\mathbb{E}[vX(v)]$:

$$\underbrace{\delta_A \mathbb{E}[z(v)]}_{\text{future utility contribution}} \geq \underbrace{u}_{\text{utility owed}} - \underbrace{(1-\delta_A)\mathbb{E}[vX(v)]}_{\text{current utility contribution}}.$$

At marginal promise cost λ, every state has the **same continuation objective**: $\delta_P \Phi(z) + \delta_A \lambda z$. It does not depend on realised v; concavity favours smoothing promises across states.

Choose a **common target** z_λ^T, unless participation requires more:

$$z_X(v;\lambda) = \max\{z_\lambda^T, \ell_X(v)\}, \qquad \ell_X(v) = X(v;u)\underline{u}(v).$$

- $\ell_X(v) \leq z_\lambda^T$: the same promise $z = z_\lambda^T$ across these states.
- $\ell_X(v) > z_\lambda^T$: participation binds; promise just enough for acceptance.

Back

Optimal Assignments are Cutoff Rules

Theorem 1(i): Assignment

There is a cutoff $\gamma : [0, V^+] \to [\underline{v}, 0]$ such that $X(v; u) = \mathbf{1}\{v \geq \gamma(u)\}$. The cutoff is constant on $[0, \bar{u}]$, nondecreasing above $\bar{u}$, and $\gamma(V^+) = 0$.

Why a cutoff? Fix the marginal promise cost λ. Compared with a lower-value task, a higher v gives:

1. the **same payoff to the firm** from completion;
2. **more current utility to the worker**, helping satisfy promise keeping;
3. a **lower participation floor**, so any continuation feasible for the lower-value task is also feasible for the higher-value task.

Not assigning gives the same continuation target z_λ^I for every v. Thus the gain from assigning is **nondecreasing in v**: if a task is worth assigning, every higher-value task is too. This gives a cutoff.

Why does the cutoff rise with u? The cutoff balances the payoff from completion against the **marginal cost of additional promises**, captured by λ. Above the plateau, higher u raises λ, making future obligations more costly.

Back

Deriving the Costate Equation

Single-agent supporting slopes

Fix state u_t, realised task value v_t, and choose $\lambda_t \in \Lambda(u_t)$. The terms in the Lagrangian that depend on the continuation promise $u_{t+1} = z(v_t; u_t)$ are

$$\delta_P \Phi(u_{t+1}) + \delta_A \lambda_t u_{t+1} + \mu_t(v_t)\{u_{t+1} - \underline{u}(v_t) X_t(v_t)\}.$$

The first-order condition for u_{t+1} is

$$\delta_A \lambda_t + \mu_t(v_t) \in \delta_P \partial(-\Phi)(u_{t+1}).$$

Pick $\lambda_{t+1} \in \Lambda(u_{t+1})$ satisfying the FOC. Then

$$\lambda_{t+1} = \frac{\delta_A}{\delta_P}\lambda_t + \frac{\mu_t(v_t)}{\delta_P} = q\lambda_t + \frac{\mu_t(v_t)}{\delta_P}.$$

- Slack participation: $\mu_t(v_t) = 0$, so the costate moves by relative patience q.
- A positive multiplier requires binding participation and raises the next costate.

◂ Slack target ◂ Back to main derivation

Strict Concavity: Intuition

Above the plateau, strict concavity means that the **marginal cost of promises rises** with the amount owed.

1. Suppose two distinct promises had the same positive marginal cost. The principal would choose the **same current assignments** under both contracts.
2. The difference in worker utility must therefore come entirely from **future promises**. Discounting shrinks the contribution of any future difference.
3. Consider the largest gap between promises with the same positive marginal cost. It must satisfy $\text{largest gap} \leq \delta_A \times \text{largest gap}$.

Promises are bounded and $\delta_A < 1$, so the largest gap is zero. Each positive marginal cost selects **one promise**, ruling out linear segments above the plateau.

▸ Proof ▸ Back

Proof: Strict Concavity

Let $A_\lambda = \arg\max_{u \in [0, V^+]} \{\Phi(u) + \lambda u\}$: the promises supported by marginal cost $\lambda > 0$. Define the largest gap $M = \sup_{\lambda > 0} \operatorname{diam}(A_\lambda) \leq V^+$.

1. **Current assignments coincide.** For $u, \widehat{u} \in A_\lambda$, the optimal cutoff is the same almost surely. Binding promise keeping gives $u - \widehat{u} = \delta_A \mathbb{E}[z(v) - \widehat{z}(v)]$.

2. **Continuation gaps are bounded by M.** The midpoint of two optimal continuations is also optimal. Its participation multiplier gives a common positive costate supporting both endpoints. Hence $|z(v) - \widehat{z}(v)| \leq M$ almost surely.

3. **Discounting closes the argument.** Thus $\operatorname{diam}(A_\lambda) \leq \delta_A M$ for every $\lambda > 0$. Taking the supremum gives $M \leq \delta_A M$, so $M = 0$.

Every positive costate selects a single promise. A linear segment above the plateau would have the same positive costate at multiple promises, so Φ is **strictly concave** there.

Back

Proof: Proposition 4

Static full assignment gives payoff 1 every period, so it is dynamically optimal whenever it is feasible.

Suppose instead $c^S > \underline{v}$. Then $c^S \in (\underline{v}, 0)$, static feasibility binds, and $U^S = \mathbb{E}[v\mathbf{1}\{v \geq c^S\}] = \underline{u}(c^S) < V^+$.

1. Pick small $\varepsilon > 0$ and lower the cutoff to $c_\varepsilon = c^S - \varepsilon$.
2. Assign all $v \geq c_\varepsilon$. For assigned negative values, set $z_\varepsilon(v) = \underline{u}(v)$; otherwise set $z_\varepsilon(v) = 0$.
3. Participation holds by construction, and promise keeping at $u = 0$ holds because positive tasks generate slack.
4. The new marginal interval has probability $m_\varepsilon = \Pr(c_\varepsilon \leq v < c^S) > 0$.
5. Continuity of Φ gives dynamic payoff at least $\Pi^S + (1 - \delta_P - \delta_P\eta)m_\varepsilon$, which exceeds Π^S for small η.

Thus, when static full assignment is infeasible, the dynamic contract strictly improves on the static optimum.

Back

Related Papers

Related Papers

- Sun and Tian (2018)
- Samuelson and Stacchetti (2017)
- Combe, Nora and Tercieux (2025)
- Forand and Zápal (2020)
- Andrews and Barron (2016)
- Bird and Frug (2019)
- Ji and Vong (2026)

Back to Related Work

Sun and Tian (2018)

Model and Results

Optimal Contract to Induce Continued Effort, Management Science.

- **Model.** A risk-neutral agent privately chooses costly effort that raises the arrival rate of Poisson successes. The principal observes successes and earns revenue from them.
- **Instruments.** The principal commits to nonnegative cash transfers and possible termination. Promised utility summarises the history.
- **Equal patience.** Promised utility reaches either zero, ending the relationship, or an absorbing payment threshold, followed by constant success bonuses.
- **Principal more patient.** Promises decline between successes and jump after successes. The payment threshold is nonabsorbing; the baseline relationship terminates in finite time almost surely.

Scope: contracts inducing high effort until termination.

Comparison Related papers

Sun and Tian (2018)

Similarities and Differences

- **Similarities.** Both use future compensation to provide incentives, and relative patience shapes the evolution of promises.
- They do not hold payment terms fixed. Cash bonuses cap continuation rewards at a payment threshold; here, a difficult job can require $z(v;u) > z^T(u)$.
- They do not have our ex post acceptance constraint for publicly observed task values. Their incentive problem is hidden effort.
- They do not derive the promise bound from future good jobs. Here, assignment values imply $u \leq V^+$.
- Their baseline $q < 1$ contract terminates almost surely. Here, promises keep fluctuating within the relationship when static full assignment is infeasible.

◂ Model ◂ Related papers

Samuelson and Stacchetti (2017)

Model and Results

Even Up: Maintaining Relationships, Journal of Economic Theory.

- **Model.** Two players repeatedly observe an i.i.d. project that benefits one player and costs the other. Both must approve; the person bearing the cost varies across projects.
- **Self-enforcement.** The baseline has public projects and actions and a common discount factor. Future cooperation must make approval optimal for both players.
- **Future projects as compensation.** Without transfers, sufficiently costly accepted projects reward the person bearing the cost through better future treatment.
- **Voluntary transfers.** Efficient relationships use continuation play that maximises total surplus. Self-enforcing payments handle costly projects and can occur infrequently.

▸ Comparison ◂ Related papers

Samuelson and Stacchetti (2017)

Similarities and Differences

- **Similarities.** In their model without transfers, future good projects compensate for accepting costly projects, as they do here.
- They do not give one player commitment power. Both must prefer to honour the relationship; here, the principal commits to future assignments.
- They do not allow different discount factors. We study all three relative-patience regimes.
- They do not study assignment across several agents. Our principal can allocate scarce projects among several agents.
- They characterise the frontier of bilateral self-enforcing payoffs. We solve the principal's optimal assignment policy with commitment and ex post participation.

◂ Model ◂ Related papers

Combe, Nora and Tercieux (2025)

Model and Results

Dynamic Assignment without Money: Optimality of Spot Mechanisms, Theoretical Economics.

- **Model.** A finite horizon and a continuum of agents; each agent must be assigned a perishable object at every date.
- **Private preferences.** Agents know their preferences over complete assignment sequences at date zero; preferences allow complementarities across dates.
- **Spot mechanism.** Each agent receives an initial random token budget, buys assignments at dated prices, and carries unused tokens forward.
- **Ordinal result.** Under full-support preferences, incentive compatibility and ordinal efficiency characterise spot mechanisms.
- **Cardinal result.** With cardinal reports and stronger regularity and support assumptions, incentive compatibility and Pareto efficiency characterise spot mechanisms with menus of initial budget lotteries.

▸ Comparison ◂ Related papers

Combe, Nora and Tercieux (2025)

Similarities and Differences

- **Similarities.** Both use accounts to link assignments over time and give agents access to future opportunities.
- They do not impose our ex post participation constraint: their baseline requires an assignment each date. Our agent can refuse or quit after observing the task value.
- They do not have unexpected preference shocks. Agents know their preferences over complete assignment paths at date zero; here, public task values are realised each period.
- They characterise efficient, incentive-compatible mechanisms. We maximise the principal's payoff subject to promise keeping and ex post participation.
- They use a finite horizon and a continuum of agents. We use an infinite horizon and finitely many agents.

Section 7 permits a null assignment and planner optimisation within the efficient class.

◂ Model ◂ Related papers

Forand and Zápal (2020)

Model and Results

Production Priorities in Dynamic Relationships, Theoretical Economics.

- **Model.** A principal commits to production decisions; an agent can quit. Public projects yield fixed utility pairs and follow an arbitrary stochastic process. Both parties share a discount factor.
- **Timing.** The agent observes the project and chooses whether to remain before a public lottery determines production.
- **Production priorities.** Demands favour high principal benefits relative to agent costs. Rewards favour high agent benefits relative to principal costs.
- **Dynamics.** Reward commitments broaden toward less cost-effective projects, while demands narrow. With finitely many project types, long-run inefficiency vanishes.

▸ Comparison ◂ Related papers

Forand and Zápal (2020)

Similarities and Differences

- **Similarities.** A committed principal uses favourable future projects to compensate current sacrifice; stochastic opportunities determine the available rewards.
- They do not require participation after the production lottery is realised. Here, the agent can reject the realised assignment.
- They do not study different discount factors. We study all three relative-patience regimes.
- They do not require every completed project to benefit the principal. Here, good jobs benefit both parties and also deliver compensation.
- They do not study competing agents. Here, scarce projects link the promises made to several workers.

The participation-timing difference matters when production is randomised.

Participation example | Model | Related papers

Forand and Zápal (2020)

Participation and the Production Lottery

A bad job costs the agent 1 unit of utility today. Promised future utility is worth $1/2$ today, whether or not the job is assigned.

- **Forand–Zápal.** After observing the job, the agent commits to a 50% chance of production. The assigned outcome gives $-\frac{1}{2}$; the unassigned outcome gives $+\frac{1}{2}$. They can **bundle good and bad outcomes** into an acceptable lottery with expected utility 0.

- **Here.** The agent can refuse the realised assignment. Every realised assignment must satisfy participation, so the agent rejects the bad outcome.

- **Our deterministic reduction.** It applies only to lotteries where **every realised assignment satisfies participation**. For a given deterministic policy, the participation constraints coincide.

◂ Comparison ◂ Related papers

Andrews and Barron (2016)

Model and Results

The Allocation of Future Business: Dynamic Relational Contracts with Multiple Agents, AER.

- **Model.** A principal selects one supplier after public productivity draws. The supplier chooses hidden effort; output and payments are observed within that relationship.
- **Relational incentives.** Cash payments are flexible and promises must be self-enforcing. Future business makes promised incentive payments credible.
- **Favoured supplier.** Among the currently most productive suppliers, choose the one who succeeded most recently. Subsequent failures by that supplier do not remove its priority.
- **Result.** Under their parameter condition, this rule achieves efficient supplier selection and high effort whenever any relational contract can. It improves on stationary rules.

Comparison Related papers

Andrews and Barron (2016)

Similarities and Differences

- **Similarities.** Future business provides incentives, and scarce assignments link several ongoing relationships.
- They do not hold cash payments fixed. Here, compensation comes through future good jobs under fixed payment terms.
- Their principal cannot commit. Future business sustains the credibility of bonuses; our principal commits to future assignments.
- They do not make every relationship's history public. Their incentive problem combines hidden effort with private bilateral histories; our task values and histories are public.
- They do not allow different discount factors. We study all three relative-patience regimes and the resulting allocation dynamics.

◂ Model ◂ Related papers

Bird and Frug (2019)

Model and Results

Dynamic Non-monetary Incentives, AEJ: Microeconomics.

- **Model.** A committed principal and one agent share a discount rate. Investment and reward opportunities arrive through separate Poisson processes.
- **Private opportunities.** The agent privately observes arrivals, while completed actions are public. Investments benefit the principal and cost the agent; rewards reverse these payoffs.
- **Identical rewards.** Optimal compensation gives the agent permission to take every reward arriving within a predetermined time window.
- **Several reward types.** The optimal mechanism uses separate time allowances for different rewards.

Comparison Related papers

Bird and Frug (2019)

Similarities and Differences

- **Similarities.** Stochastic opportunities limit compensation; future access to rewards induces costly current actions.
- Their baseline does not make opportunity arrivals public. Our task values are publicly observed, and acceptance follows each realised assignment.
- They do not use one task stream that always benefits the principal. Their investments and rewards have opposite payoff effects; here, good jobs benefit both parties.
- They do not study different discount rates. We study all three relative-patience regimes.
- They study one agent. Our principal can allocate several simultaneous projects among several agents.

Section IV.B also studies public investment opportunities and the agent's ability to refuse.

◂ Model ◂ Related papers

Ji and Vong (2026)

Model and Results

Task Assignment with Interdependent Incentives, September 2026 working paper.

- **Model.** A manager assigns one burden among identical workers. Effort is private; success or failure is public.
- **Instruments.** Assignment provides incentives. The baseline has no cash, no managerial commitment, and a common discount factor.
- **Optimal priority.** Among equilibria inducing effort whenever assigned, retain the worker after failure and rotate after success. Vacancy phases strengthen incentives when necessary.
- **Reward and inequality.** Success earns relief until other workers succeed and the vacancy phase ends. Queue positions give identical workers different expected discounted future workloads.

▸ Comparison ◂ Related papers

Ji and Vong (2026)

Similarities and Differences

- **Similarities.** Future assignment provides incentives, and rewarding one worker changes the opportunities available to others.
- They induce hidden effort. We induce acceptance after publicly observed task values.
- Their baseline manager cannot commit. Our principal commits to future assignments.
- Their baseline has a burden task and common discounting. We allow good and bad jobs within the same relationship and different discount factors.
- They do not analyse multiple simultaneous tasks in this version. We allow up to K projects to be assigned among $N \geq K$ agents.

Section 6 adds participation constraints, desirable tasks, and other extensions.

◂ Model ◂ Related papers