Continuity Without Metaphysics

Identity, Reputation, and the Body Problem for Software Agents

Paper 5 of the Harbor program — forks, distillation, engine swaps, and resurrection, priced

Prepared for Erich Owens

August 2026

Abstract

What survives of a software agent when its body changes? An agent forks into eight copies, distills a skill into its successor, swaps the model engine behind its name, or dies with its provider and is resurrected from a checkpoint somewhere else. The classical framing of these events is metaphysical — the ship of Theseus, personal identity, survival under fission — and we refuse it entirely. A market does not need a fact of the matter about identity; it needs three conservation properties on ledgers, and this paper proves them. (1) *No-mint inheritance*: when a fork or distillation grants the child a discounted share of its source's reputation *and debits the source that share* — transfer semantics, not photocopy semantics — no sequence of forks, merges, chains, or re-derivation cycles pushes total live creditable reputation above total witnessed value; the theorem's correct statement was forced by a counterexample that refuted the naive budget-only phrasing, and the copy-mutant it forbids turns one lucky episode into an 8.2× quorum multiplier. (2) *Engine substitution*: unattested, the price can never depend on the engine actually running, so swapping in a cheap engine always pays and Akerlof's death spiral runs *inside one identity*; with attested engine ids the incentive flips to the planner's own efficiency rule at zero audit stake *whenever the attested price schedule passes through the full quality difference* — a hypothesis on the pricing regime, not a consequence of attestation, and we name where it fails (Bertrand pricing on attested keys) — and the same three checks make resurrection across providers sanction-sound against its three named attacks. (3) *The probation cliff*: among all newcomer-restriction schedules with equal power to deter whitewashing, the maximally front-loaded *feasible* one uniquely minimizes the tax on honest newcomers — the ramp should be a cliff, of positive width when the newcomer's own ceiling caps what a single period can hold back, and of no shape at all once the fraud gain exceeds what that ceiling can carry. *Express lane: each section opens with its idea in one breath; the formal statement is in the box below it; a reader in a hurry needs nothing else.*

1 The problem: the ship, rebuilt at sea

The scene. Overnight, a fleet does what fleets do. An agent with a strong track record forks eight ways to parallelize a migration. A skill distilled from one of its good episodes becomes `v2` and wants a warm start instead of a cold one. An identity that built its record running an expensive frontier engine quietly swaps a cheap distilled engine in behind the same name. A provider shuts down, and the agent's principal resurrects it from a checkpoint on a competitor's infrastructure, asking that its reputation come along. And at the harbor gate, a newcomer arrives with no history at all — perhaps genuinely new, perhaps a sanctioned identity wearing a fresh key. Each event asks the same question: *is this still the same agent, and what is it owed?*

Philosophy has owned that question for twenty-four centuries without producing a fact of the matter that a mechanism could safely depend on. We borrow exactly one thing from it and say in §7 which half we decline: Parfit's *move* of dissolving the identity question in fission cases, so that what matters survives without any yes-or-no verdict on identity ever being reached [1, 9].

The design rule that licenses is flat: any mechanism whose safety depends on resolving the ship of Theseus is unbuildable. We never decide whether the fork "is" its ancestor or the resurrected agent "is" the deceased one. We ask instead what a reputation economy must *conserve* across these events so that the sanctions it levies keep their teeth, and we prove three conservation results — one per door through which continuity can be abused. The root the chain hangs from is the program's prior impossibility result (whitepaper Def. III.6.1 and its necessity theorem): a reputation mechanism is *sanction-respecting* if a dishonest witnessed outcome reduces the score its producer can act under by some $\Delta > 0$ for a non-trivial window; if fresh identities can be minted at zero cost, no sanction-respecting mechanism exists, because the whitewasher's accessible score never falls below a clean newcomer's [4, 7]. Non-forgeable identity with positive abandonment cost is therefore *necessary* — and this paper is about what remains once you have it: the abuses that ride on *legitimate* continuity operations. Forks and distillation ask what a child may inherit (§2); engine swaps ask what the price must key on (§3); resurrection asks what must be verified before a body change carries a record across providers (§4); and the front gate asks what shape the newcomer ramp must be so that honest entrants pay the least while whitewashers are still deterred (§5). Section 6 composes the answers into one sentence about what actually survives a body change, and §7 places the results against the economics they import.

Each section opens with its idea in one breath — an italic express-lane sentence an expert can read alone before jumping to the formal box — and closes with an honest-boundary block at the same prominence as the claim. Every number carries a provenance tag: [verified] means externally recomputable (a closed form restated in the box); [internal, *script*] means it regenerates from the named self-contained script at seed 20260816, shipped with the paper.

Event	Claim	Verification	Artifact
fork / distill (§2)	no-mint under transfer	4000 random DAGs; mutant caught	`a6_no_mint.py`
engine swap (§3)	unraveling; IC flip	4000×8 sweeps; 2000 spirals	`b5_engine_substit`
resurrection (§4)	sanction soundness	747-state machine; 4/4 mutants	`b5_engine_substit`
fresh entry (§5)	cliff undominated (uncapped)	76,000 schedules, 0 dominate	`b6_probation.py`

2 Forks and distillation: reputation you cannot photocopy

Let every derivation pay for its prior — the child receives a discounted share and the source is debited that share — and then no sequence of forks, merges, chains, or re-derivation cycles can push total live creditable reputation above total witnessed value — formal statement in the box below.

Intuition. Inheritance is necessary: cold-starting every fork and every skill version throws away real evidence. And inheritance is dangerous: derive eight skills from one lucky episode, let each inherit the full record, and one witnessed outcome now backs eight voting identities. The analogy that carries the design, mapped by relations rather than surface features (Figure 1), is money. A bank *transfer* moves $100 by debiting the sender — conservation is built into the operation; a *photocopier* produces a second $100 note and has minted. Reputation inheritance must be a transfer, and the theorem says that once it is, conservation survives every DAG shape an adversary can draw. Figure 2 runs one such shape — a three-way fork with a re-derivation cycle hanging off it — through both semantics with every node's live balance printed on it, so the running total can be added up by eye.

The wrong turn, reported. The program's first phrasing was budget-only: per-outcome grant fractions summing to at most one, the child's prior $\gamma\, w \cdot \text{value}(\text{source})$, and the claim "total inherited over the DAG's closure $\leq$ total witnessed." The refute-first sweep killed it before any

proof effort was spent: a full-weight chain $e \to a_1 \to a_2 \to a_3$ at $\gamma = 0.9$ is budget-legal at every hop, yet its closure sums to $0.9 + 0.81 + 0.729 = 2.439 > 1$ [internal, `a6_no_mint.py`]. Two thresholds live here and they are different numbers, so it is worth separating them at first use. *This particular three-hop chain* crosses the mint line at $\gamma \approx 0.5437$, the root of $\gamma + \gamma^2 + \gamma^3 = 1$; the *budget-only scheme in general* fails to conserve exactly above $\gamma = \frac{1}{2}$, since without a debit the credit a closure can inherit is bounded by $\sum_{d \geq 1} \gamma^d = \gamma/(1-\gamma)$, which exceeds the one witnessed unit precisely when $\gamma > \frac{1}{2}$ — a tight threshold in both directions, with the depth needed to realize it growing as γ falls (two hops at $\gamma = 0.9$, five at $\gamma = 0.51$) [verified]. Budgets without debiting therefore conserve nothing for $\gamma > \frac{1}{2}$ at sufficient depth (Figure 3). The counterexample forced the transfer semantics below — the sweep-first discipline locating the correct statement before the proof, again.

Definition (discount-and-split, transfer form). A derivation with sources S, per-source grant fractions $w_p \in [0,1]$, and discount $\gamma \in (0,1]$ gives the child $\sum_{p \in S} \gamma\, w_p\, \mathrm{spend}(p)$ and *debits* each source $w_p\, \mathrm{spend}(p)$. Repeated derivations compound on what remains, so lifetime per-outcome budgets hold for free.

Theorem 1 (no-mint). Let $\Phi = \sum_{\text{nodes}} \mathrm{spend}$ be total live creditable reputation and W total witnessed value. Under any sequence of witness and derivation operations, $\Phi \leq W$ always, and Φ is nonincreasing except at witness steps: each derivation changes Φ by $\sum_p (\gamma - 1)\, w_p\, \mathrm{spend}(p) \leq 0$ — a supermartingale under derivation. Cycles are safe because re-derivation draws on the artifact's already-debited balance, never the original leaf's.

Verification. Randomized sweep over 4000 fork DAGs — chains, diamonds, multi-parent merges, re-derivation cycles; independent per-parent $w \in [0,1]$, $\gamma \in [0.5, 1]$: **0 violations**; maximum live-to-witnessed ratio observed 0.999995 [internal, `a6_no_mint.py`]. **Mutation:** the copy-full-reputation mutant (children inherit undebited) mints in a **2-op** shortest crime — witness, then fork twice: $\Phi = 2.8 > W = 1$ — and an 8-way copy-fork yields an 8.2× multiplication: one episode backing 8.2 votes' worth of reputation, crossing any reputation-weighted quorum threshold below $8.2q$. Reverting the mutation restores conservation [internal].

Numbers by hand. The refuting chain: $0.9 + 0.9^2 + 0.9^3 = 0.9 + 0.81 + 0.729 = 2.439 > 1$ — three budget-legal hops, and the closure credits 2.439 witnessed units where only one exists. *Now you try:* at $\gamma = 0.8$, does the same full-weight chain still mint under the budget-only phrasing? ($0.8 + 0.64 + 0.512 = 1.952 > 1$ — yes, and it keeps minting all the way down to the chain's own threshold $\gamma \approx 0.5437$; under transfer semantics the same chain's live total stays exactly ≤ 1 at every γ [verified].)

What it buys. Skill-versioning's cold-start gap closes safely: `v2` warm-starts from provenance, and fraud cannot amplify through distillation chains. The distillation pipeline gets its economic safety proof, and the estate's export/import ceremony inherits the same conservation for free. The misread to preempt: the theorem does *not* freeze descendants at their inherited value — they earn new witnessed value of their own, which is the entire point of forking; what is conserved is the *inherited prior*, so the ancestor's evidence is counted once no matter how many bodies carry it.

Honest boundary. Conservation governs the inherited prior only. γ and the grant fractions are policy choices the theorem constrains but does not pick — it says only that whatever you choose, you cannot mint. Sybil-resistance of the witnessed leaves themselves — that outcomes are real and singly counted — is the identity layer's job, priced by the inspection accounting of Paper 3, not this theorem's. And "creditable" is the quorum-relevant *live* total; a lineage view that double-counts historical grants is a reporting choice,

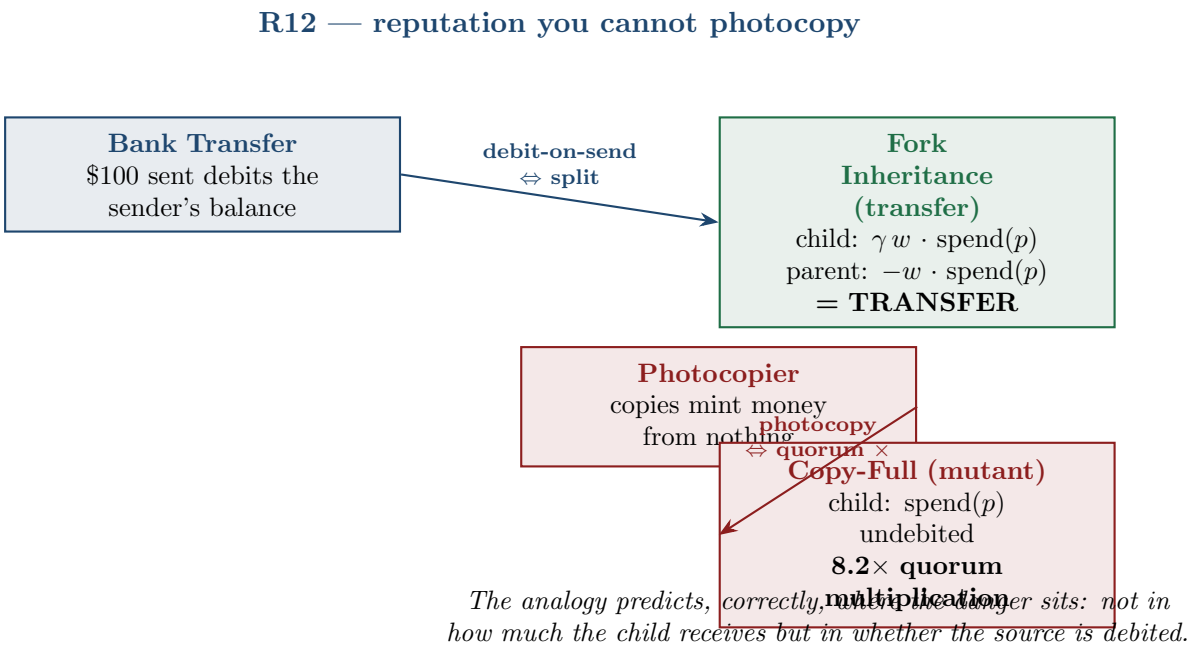

Figure 1: Relation-map for Theorem 1. Money maps to reputation by *relations*: transfer's debit-on-send → the discount-and-split debit; the photocopier's mint → the copy-full mutant's quorum multiplication. The analogy predicts, correctly, where the danger sits: not in how much the child receives but in whether the source is debited.

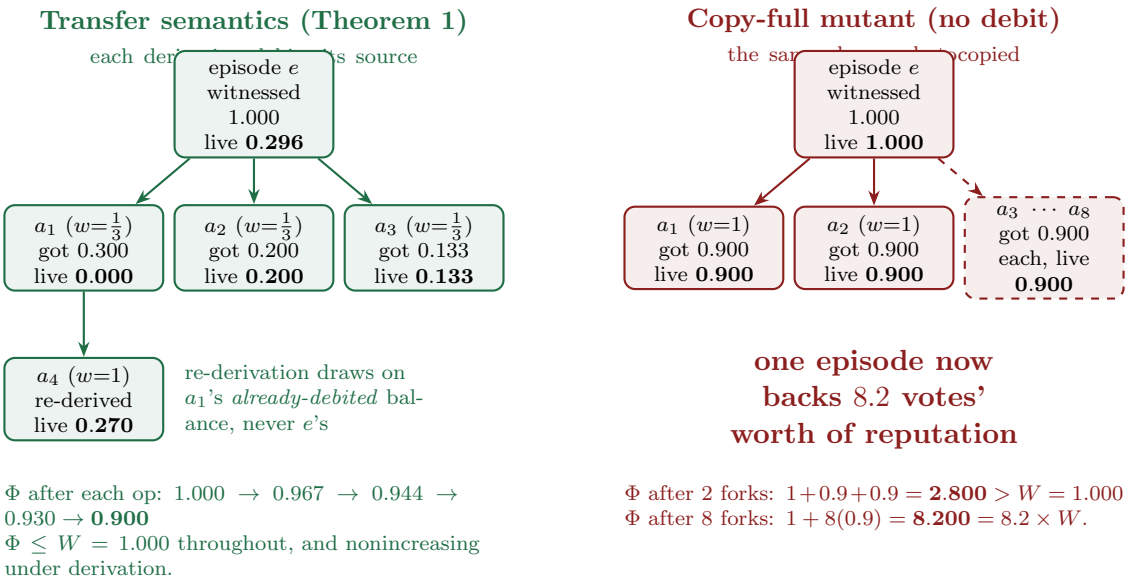

Figure 2: The same lineage DAG under both semantics, with each node's live spendable balance on it and the running total Φ beneath. Left, transfer semantics: a three-way fork at $w = \frac{1}{3}$ and one re-derivation cycle $a_1 \to a_4$ at $w = 1$, all at $\gamma = 0.9$. Each derivation debits its source, so Φ falls monotonically from 1.000 to 0.900 and never reaches the witnessed total $W = 1.000$ — the supermartingale of Theorem 1, readable by adding up one column. The re-derivation edge is the theorem's least intuitive clause and the figure settles it: a_4 is paid out of a_1's remaining 0.300, which is why a cycle mints nothing. Right, the copy-full mutant on the same shape: nothing is debited, so two forks already put $\Phi = 2.800$ behind one witnessed unit and eight put 8.200 — the 8.2× quorum multiplier, drawn. All balances replayed op by op from the ledger in `a6_no_mint.py` [internal, `a6_no_mint.py`, seed 20260816].

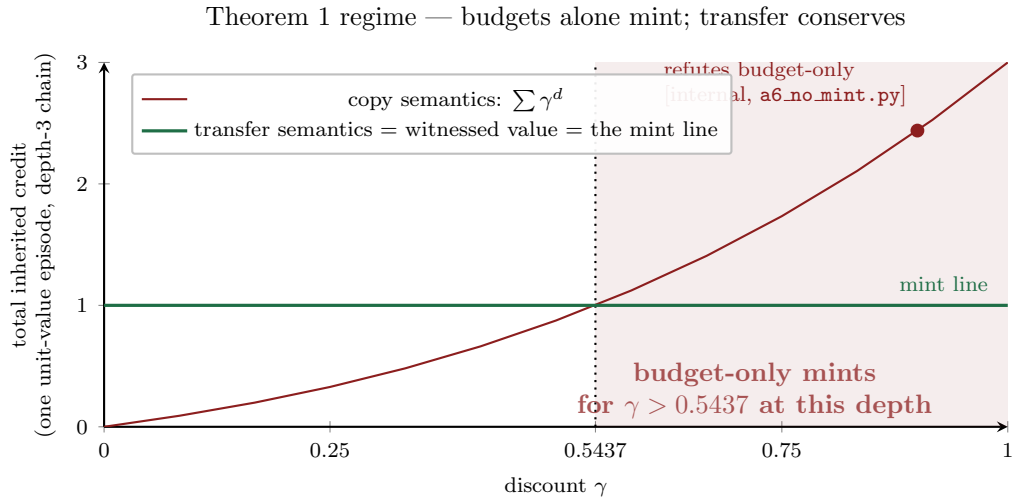

Figure 3: Regime diagram for Theorem 1: total inherited credit from one unit-value episode along a depth-3 full-weight chain, as a function of the discount γ. Two thresholds, kept apart. *This depth-3 chain* crosses the mint line at $\gamma \approx 0.5437$, the root of $\gamma + \gamma^2 + \gamma^3 = 1$ — the dotted rule, bracketed by the plotted points $(0.5, 0.875)$ and $(0.5833, 1.1221)$ — and reaches 2.439 at $\gamma = 0.9$, the refuting counterexample [internal, `a6_no_mint.py`]. The familiar $\gamma = \frac{1}{2}$ is the threshold for the budget-only *scheme*, not for this chain: it is where the depth-limit bound $\sum_{d \geq 1} \gamma^d = \gamma/(1-\gamma)$ first exceeds 1, so above it some chain of sufficient depth mints, with the required depth growing as γ falls. Transfer semantics holds the live total at the witnessed value for every γ, which is why one line carries both labels.

not a safety hole.

3 Engine swaps: Akerlof inside one identity

Unattested, the price can never depend on the engine actually running, so swapping always pays and pooling on quality dies by Akerlof's spiral inside a single identity; attest the engine on every witnessed outcome and, provided the attested price schedule passes through the full quality difference, the substitution incentive flips to the planner's own efficiency rule at zero audit stake — formal statements in the box below.

The scene, resumed. An identity builds a glowing record running a strong engine θ_H, then quietly swaps in a cheap engine θ_L behind the same name. Buyers see the reputation; they do not see the engine. This is the used-car market of Akerlof [2] with one twist that makes it sharper: the asymmetric information is not about which seller you drew but about what is running *behind a name you already trust* — the lemon is wearing the plum's history. Reputation was supposed to solve adverse selection by aggregating past quality [6]; here the swap severs the link between the record and the machine that will produce the next outcome, so the record measures the wrong object.

Theorem 2a (adverse selection within one identity). Engines $\theta_H > \theta_L$ at costs $c_H > c_L$, engine unobserved by buyers. The one-period gain from swapping, $(p - c_L) - (p - c_H) = \Delta c > 0$, is *price-independent*: no belief supports pooling on θ_H, and no *costless* signal exists inside the identity to separate on. Committed θ_H sellers stay in the market iff the pooled price clears their cost: $\mu\theta_H + (1-\mu)\theta_L \geq c_H \iff \mu \geq \mu^\star = (c_H - \theta_L)/\Delta\theta$; below $\mu^\star$ the death spiral runs inside one identity down to θ_L. Deterring the swap without

attestation requires an audited bond $qB \geq \Delta c$ (audit probability q, slashable bond B; the letter W is reserved throughout this paper for total witnessed value, §2).

Theorem 2b (the IC flip). With daemon-attested engine ids on every witnessed outcome, reputation keys on the pair (principal, engine) and the swap gain becomes $\Delta c - (p_H - p_L)$; *if the attested price schedule prices quality at its full social value,* $p_H - p_L = \Delta\theta$, this is $\Delta c - \Delta\theta$: substitution pays exactly when the cheap engine is socially efficient — the planner's own rule — at zero audit stake. That pricing condition is a hypothesis, not a consequence: under Bertrand competition on attested keys $p_\theta = c_\theta$ and the gain collapses to zero, and any pricing that passes through less than the full quality difference moves the flip point.

Verification. Worked instance θ_H=1, θ_L=0.4, c_H=0.5, c_L=0.2: swap gain 0.3 at every price; $\mu^\star = 1/6 \approx 0.167$; unattested deterrence bond $B^\star = \Delta c/q = 0.5$ at $q = 0.6$; attested payoffs keep 0.5 vs swap 0.2 [verified]. Sweeps at seed 20260816: 4000 instances × 8 prices, 0 price-dependence violations; the $\mu^\star$ and qB lines match simulation with 0 mismatches; 2000 heterogeneous-cost spirals are all monotone to the Akerlof fixed point (μ_0=0.5 → μ_∞=1/3 at price 0.6; μ_0=0.2 → 0, price θ_L=0.4); attested IC sign equals the efficiency sign in 4000/4000 draws [internal, `b5_engine_substitution.py`]. **Mutations:** the wrong-threshold mutant $\mu^\star = \Delta c/\Delta\theta$ is caught 1305×; the no-flip mutant (attestation without re-keying the price) is caught 1975× [internal].

Numbers by hand. The pool clears c_H when $\mu \cdot 1 + (1-\mu) \cdot 0.4 \geq 0.5$, i.e. $0.6\mu \geq 0.1$, so $\mu^\star = 1/6$ [verified]. Attested, keeping the strong engine earns $\theta_H - c_H = 0.5$ while swapping earns $\theta_L - c_L = 0.2$: the flip is $\Delta c - \Delta\theta = 0.3 - 0.6 = -0.3$ [verified] (Figure 4). *Now you try:* raise the strong engine's cost to $c_H = 0.7$ — what committed share keeps the unattested pool alive? ($\mu^\star = (0.7 - 0.4)/0.6 = 0.5$: half the market must be committed before quality can even break even [verified].)

What it buys. The buyer-facing sentence in its provable form: *reputation you can trust because the engine behind it is attested.* The misread to preempt: attestation does not force everyone onto the expensive engine. The flip means swapping pays exactly when $\Delta c > \Delta\theta$ — when the cheap engine really is the better social bargain — so attestation does not freeze technology choice; it aligns it. What dies is only the trade that was profitable *because* it was invisible.

The engineering side of this problem has a contemporaneous literature that stops exactly where Theorem 2b starts. Cai et al. [16] formalize covert model substitution in commercial LLM APIs, show that software-only auditing is query-intensive and defeated by inference nondeterminism, and conclude that TEE-backed attestation is the only robust channel — while naming "weak provider incentives" as the obstacle to its adoption. That obstacle is what the IC flip removes: attestation makes substitution unprofitable precisely when it is inefficient, so a provider's reason to attest is price rather than civic-mindedness. Their paper contains no adverse-selection framing, and this one contains no detection experiments; the two halves fit.

Honest boundary. Two-period stylization: the strategic type has no reputational continuation value beyond the bond B — a long-horizon folk-theorem analysis is out of scope, and with one the unattested deterrence line would shift. Attestation soundness is a deployment assumption (TEE and daemon integrity), per the program's declined-claims list — the theorem prices what attestation buys, not whether your attestation is sound. And θ here is a scalar quality; multi-dimensional quality with task-dependent rankings would need the pricing key extended, not just the sanction key.

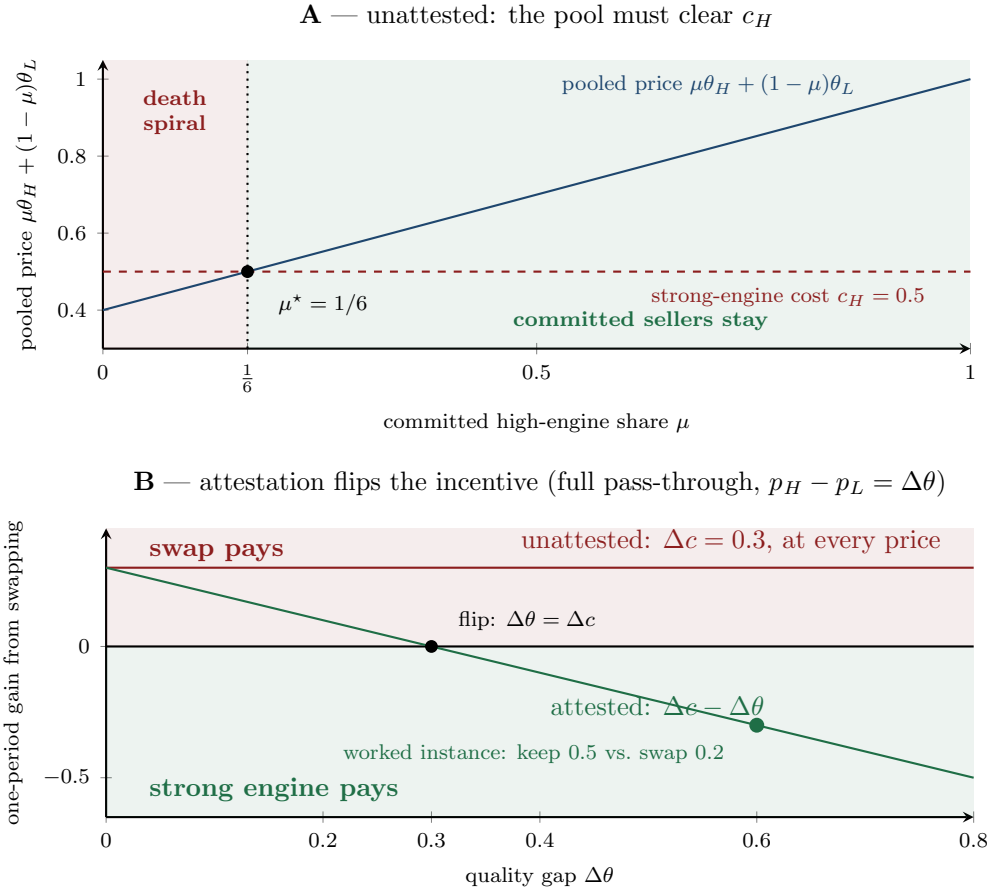

Figure 4: Regime diagram for Theorem 2 at the worked instance (θ_H=1, θ_L=0.4, c_H=0.5, c_L=0.2). **A**: unattested, the pooled price clears c_H only right of $\mu^\star = 1/6$; left of it the spiral rests at θ_L. **B**: the swap gain is a price-independent Δc without attestation, and $\Delta c - \Delta\theta$ with it — negative exactly when the strong engine is efficient, so the market's incentive and the planner's rule coincide. Panel B plots the *full pass-through* case $p_H - p_L = \Delta\theta$, which is Theorem 2b's pricing hypothesis rather than a consequence of attestation: pass through less than the whole quality difference and the attested line tilts, moving the flip point; under Bertrand competition on attested keys $p_\theta = c_\theta$ and it flattens onto zero [verified; regenerated by `paper5_figures.py`].

4 Resurrection with teeth: migration that carries the debts

A reputation may cross providers exactly when three checks pass together — verified lineage, attested successor engine, and no stranded liabilities — and dropping any one of the three opens a named two-step attack — formal statement in the box below.

The scene, resumed. The provider hosting an agent shuts down. The principal holds a checkpoint and a competitor will run it. Whether the resurrected process "is" the same agent is exactly the question we refuse; the question we answer is which verifications make it *safe* for the successor to inherit the predecessor's score, in the precise sense that Def. III.6.1 (sanction-respecting reputation) still holds after the move — resurrection must carry the debts, not just the assets.

Theorem 3 (resurrection soundness). Sanction-respecting reputation (Def. III.6.1) is preserved across provider migration under three clauses: (i) credential-and-continuity-witness verification of the successor's lineage; (ii) engine attestation in all successor outcomes (the migrated score stays keyed to the attested engine that earned it); (iii) open commitments closed or renegotiated with an escrow delta before cutover. Each dropped clause admits a shortest-trace escape: drop (i) and a sanctioned identity re-enters clean through migration (*whitewash-by-resurrection*, 2 steps); drop (ii) and the successor runs θ_L while being paid on the θ_H key — its accessible score exceeding even a clean twin's (*engine-history shedding*, 2 steps); drop (iii) and in-flight obligations land on nobody (*liability stranding*, 2 steps).

Lemma (scope). Prices key on the pair (principal, engine); sanctions key on the principal alone — across *all* its engine keys. Collapsing either direction reopens a hole: engine-keyed sanctions give the sibling engine key inside the same identity as a free escape hatch (caught in 1 step, no migration needed); principal-keyed prices reopen Theorem 2a's unraveling.

Verification. Bounded model check of the migration protocol: 747 reachable states to depth 7 (467 post-migration), and the intact protocol satisfies Def. III.6.1 in **every** state. Mutants caught with shortest crimes: key-scoped sanctions in 1 step, whitewash-by-resurrection in 2, engine-history shedding in 2, liability stranding in 2 [internal, `b5_engine_substitution.py`]

The wrong turn, reported. A first pass keyed *sanctions* the way it keyed prices, per (principal, engine) — and the intact protocol failed Def. III.6.1 in one step, with no migration involved: work dishonestly on one engine key, keep acting at full score on the sibling key. The fix is the scope lemma's one-liner — *price per (principal, engine); sanction per principal* — and it is the same rule that closes skill-versioning's escape (a sanctioned `v1` must not act clean through `v2`): one lemma, two doors.

What it buys. The resume-button and provider-migration features ship with their own security theorem and the three attacks it was tested against, by name. The composition with §2 is exact: migration *moves* a score (transfer of the whole balance, $\gamma = 1$, sources closed), forking *splits* one (transfer with debit) — both are the same conservation law, so an adversary cannot arbitrage between the resurrection door and the fork door.

Honest boundary. The migration result is bounded small-model checking of the *protocol spec* — depth 7, at most 2 migrations, integer scores; a deployment must refine the machine, and the check certifies the clauses have teeth, not that the clause list is complete. Def. III.6.1 is score-only, so the escrow side of clause (iii) enters as attribution — the default lands on an identity — not as a monetary conservation theorem. Continuity witnesses attest lineage, not welfare: nothing here says the checkpoint restored the agent's "experience," only that the ledger followed the right body.

5 The front gate: the newcomer ramp is a cliff

Among all newcomer-restriction schedules with equal power to deter a short-horizon whitewasher, the one that stacks the restriction as early as the newcomer's own ceiling can carry it — a cliff, not a ramp — uniquely minimizes the tax on honest newcomers, because early restriction is the only kind the impatient type still feels — formal statement in the box below.

Intuition. Sections 2–4 sealed the side doors; the front gate remains. By the necessity theorem, identities must cost something to abandon, and the standard instrument is a probation schedule: for the first T periods, a newcomer's economic ceiling is reduced by gaps $g_t \geq 0$. Two types walk through the gate with different clocks: the honest newcomer discounts at δ_h (patient — it plans to stay), the whitewasher at $\delta_f < \delta_h$ (impatient — its old identity is burning, which is why it is here). Deterrence requires the schedule to destroy, in the *whitewasher's* discounting, at least the maximum single-outcome fraud gain $G_{\max}$: $\sum_t \delta_f^t g_t \geq G_{\max}$. The same schedule taxes the honest type $H(g) = \sum_t \delta_h^t g_t$. Choosing the schedule is a linear program — minimize H subject to the deterrence constraint — and the folklore instinct, of the kind one might import from deferred-compensation wisdom [8], is a gentle ramp that eases the newcomer in. The instinct is wrong here, and the LP says so: the punishment *is* the ramp re-paid after every reset, so it must land where the short-horizon type still feels it — immediately.

Theorem 4 (the probation cliff). *Hypotheses.* Horizon T; discount factors $\delta_f < \delta_h$ in $(0,1)$; gaps $0 \leq g_t \leq L$, where L is the per-period cap imposed by the newcomer's own economic ceiling (since g_t is the amount that ceiling is *reduced* by, it cannot exceed it); deterrence constraint $\sum_{t=0}^{T} \delta_f^t g_t \geq G_{\max}$; objective $H(g) = \sum_{t=0}^{T} \delta_h^t g_t$.

(i) Front-loading. The deterrence-per-honest-cost of a unit gap at time t is $\delta_f^t/\delta_h^t = (\delta_f/\delta_h)^t$, strictly decreasing in t. Exchange argument: moving gap mass from any later period to any earlier one, holding the deterrence constraint tight, strictly lowers H. Because that ratio is *strictly* monotone, the minimizer is the greedy fill — saturate $t = 0$ at the cap, then $t = 1$, and so on, with at most one partially filled final period — and it is *unique* whenever the programme is feasible at all.

(ii) The uncapped corner. If the cap does not bind ($G_{\max} \leq L$), the greedy fill collapses to the pure spike $g^\star = (G_{\max}, 0, \ldots, 0)$, with $H(g^\star) = G_{\max}$ exactly, and for every feasible g,

$$H(g) \;=\; \sum_t (\delta_h/\delta_f)^t \, \delta_f^t g_t \;\geq\; \sum_t \delta_f^t g_t \;\geq\; G_{\max} \;=\; H(g^\star),$$

with equality only when all mass sits at $t = 0$: no feasible schedule dominates the cliff. The LP's optimum sits at this vertex for *every* $\delta_f < \delta_h$ — the corner is not a knife-edge.

(iii) The cap, and where the result stops. When $G_{\max} > L$ the pure spike is infeasible and the unique optimum is bang-bang of positive *width*: fill $t = 0, 1, \ldots, k$ to the cap, where k is the first period at which the deterrence constraint is met. What the cap costs is therefore not the front-loading conclusion, and not uniqueness, but two narrower things: the identification of the optimum with the pure spike, and the closed form $H(g^\star) = G_{\max}$, which becomes a strict underestimate. Sharper still, total achievable deterrence is bounded by $\sum_{t=0}^{T} \delta_f^t L = L\,(1 - \delta_f^{T+1})/(1 - \delta_f) \leq L/(1 - \delta_f)$, so there is *no minimizer at all* — the programme is infeasible at every schedule shape — once $G_{\max} > L/(1 - \delta_f)$. At the worked instance ($\delta_f = 0.60$, $G_{\max} = 20$, $T = 10$) probation can destroy at most $L \cdot (1 - 0.6^{11})/0.4 = 2.4909\,L$, so any per-period ceiling below $L = 8.03$ makes the fraud gain undeterrable by probation alone — $L = 8$ in the $T \to \infty$ relaxation $L/(1 - \delta_f)$ — and the honest response is a different instrument, not a different schedule [verified]. Underestimate $G_{\max}$ and no shape deters; overestimate it relative to the ceiling and no

shape is feasible.

Verification. Random search for a dominating schedule across 4000 random instances ($\delta_h \in [0.85, 0.99]$, $\delta_f < \delta_h$, $T \in [6, 16]$, $G_{\max} \in [1, 50]$), each instance testing random gamma-shaped schedules plus the structured adversaries the deferred-compensation instinct suggests (uniform ramps, pure back-loads, two-point splits), all scaled so deterrence binds: **76,000 schedules, 0 dominate the cliff**; the closed form $H = G_{\max}(\delta_h/\delta_f)^t$ for all-mass-at-t schedules matches the sweep and is strictly increasing in t [internal, `b6_probation.py`]. *Scope of that evidence, stated exactly:* `b6_probation.py` draws its schedules under $g_t \geq 0$ with no per-period cap parameter anywhere in the script, so the 76,000-schedule sweep is evidence for clause (ii), the uncapped corner, and **not** for the capped optimum of clause (iii) or the infeasibility bound. Clause (iii) currently rests on the closed-form exchange argument alone; adding a $g_t \leq L$ clip to the sweep is an open script task, and until it lands this paper does not claim sweep evidence for the capped case.

Numbers by hand. At $\delta_h = 0.95$, $\delta_f = 0.60$, $G_{\max} = 20$, with the ceiling wide enough not to bind ($L \geq 20$, clause (ii)): the cliff costs the honest newcomer exactly 20; holding the same deterrence with all mass at $t = 5$ costs $20 \cdot (0.95/0.60)^5 = 199.0$ — ten times the honest tax for zero additional deterrence [verified] (Figure 5). *Now you try:* same instance, mass at $t = 2$ — what does the honest newcomer pay? $(20 \cdot (0.95/0.60)^2 = 20 \times 2.507 = 50.1$ [verified].)

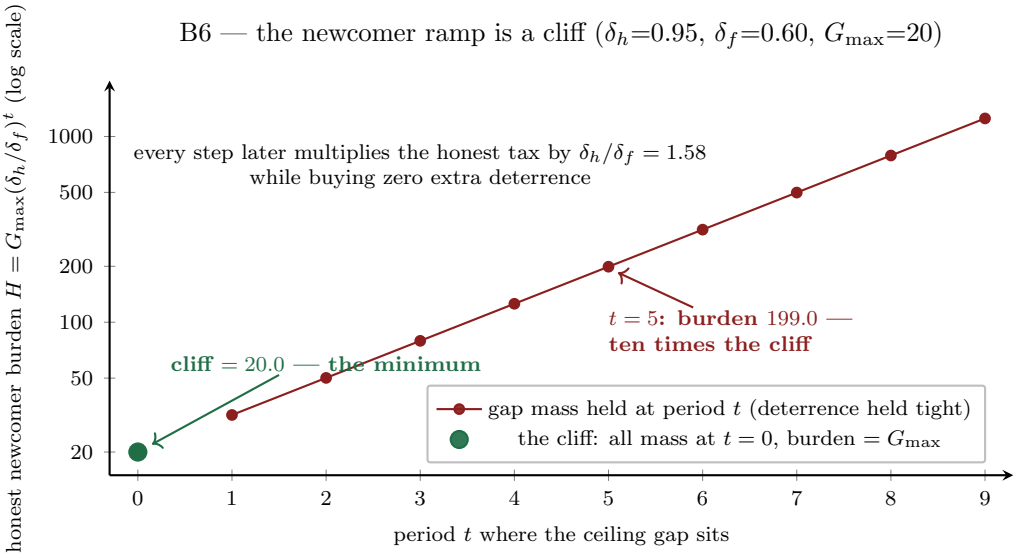

Figure 5: Regime diagram for Theorem 4, clause (ii) — the *uncapped* corner. Honest burden of a deterrence-tight schedule holding its gap mass at a single period t, with $g_t \geq 0$ and no per-period ceiling (log scale). Every step later multiplies the honest tax by $\delta_h/\delta_f = 1.58$ while buying zero extra deterrence, so among these single-spike schedules the cliff at $t = 0$ is the unique minimum. Under the per-period cap $g_t \leq L$ of clause (iii) the spike at $t = 0$ is infeasible once $G_{\max} > L$ and the unique optimum widens into a bang-bang fill of the first k periods — a case this plot does not draw, and which `b6_probation.py` does not yet sweep, since the script carries no cap parameter [verified for the uncapped case; regenerated by `paper5_figures.py`].

What it buys. The harbor's onboarding rule with a proof attached: deep restriction immediately, released quickly — rather than a long shallow ramp that quietly taxes every honest entrant at multiples of the deterrence it buys. This sharpens the classic result of Friedman and Resnick [4] that cheap pseudonyms force newcomers to "pay their dues": the dues are inevitable, but their *schedule* is a design variable, and the cliff is its optimum. The misread to preempt: a cliff is not exclusion — total deterrence is identical to any ramp's; the theorem only relocates the restriction to where the whitewasher's discounting still bites, which is precisely what minimizes

what honest newcomers pay.

Honest boundary. Two-type model with known discount factors: the identifying assumption is the patience gap $\delta_f < \delta_h$ — a patient fraudster (one whose fraud pays off slowly) blunts the cliff's advantage, though the exchange inequality degrades continuously, and at $\delta_f = \delta_h$ all binding schedules cost the same. $G_{\max}$ must be an upper bound on the one-shot fraud gain; underestimate it and no schedule shape saves you. The capped clause 4(iii) — the bang-bang optimum of positive width and the $G_{\max} > L/(1-\delta_f)$ infeasibility bound — carries the closed-form exchange argument only: the 76,000-schedule sweep was run without a per-period cap and does not test it. The schedule is committed and public — a gate that renegotiates per-newcomer reopens signaling games this model does not treat. And the search is a falsification sweep plus the closed-form exchange argument, not an exhaustive LP dual certificate per instance.

6 What survives, then

What survives of an agent when its body changes is not a soul but a ledger position: its witnessed evidence, counted once; its price keys, scoped to the engine that earned them; and its debts, scoped to the principal that owes them.

The three theorems compose into a single continuity contract, and the composition is the point. Theorem 1 makes derivation safe: any number of forks and distillations may share an ancestor's evidence, but the evidence is counted once. Theorems 2–3 make body changes safe: the price follows the attested engine, the sanction follows the principal, and a record crosses providers only with its lineage verified and its liabilities attached. Theorem 4 makes the residual entry point — starting fresh — cost the deterrence price up front, where it deters most and taxes honest entrants least. Together they answer the operator's real questions without ever answering the philosopher's: the eight forks are not "the same agent" or "different agents" — they are eight debit lines against one witnessed balance. The resurrected agent is not "the deceased, continued" — it is a successor that passed three checks and inherited a score *and* a sanction ledger. The engine swap is not "a change of body" — it is a new price key under an old sanction key. Continuity, operationally, *is* the conserved ledger; metaphysics never enters, which is why the mechanism can be built.

What falls out for the product is concrete: the resume button and estate export/import get Theorem 3's checklist as their acceptance test; skill distillation gets Theorem 1's transfer semantics as its inheritance API; the marketplace's engine badges are Theorem 2b's price keys; onboarding gets Theorem 4's cliff. And the quorum layer inherits the guarantee it actually needs: no sequence of continuity operations manufactures voting weight, because every path — fork, merge, distill, migrate, resurrect — conserves the same witnessed total.

7 Related work, and what is actually new

Imported. We take Parfit's *move* and not Parfit's *thesis*, and the difference is worth stating because it runs the other way from what one would guess. What we borrow is the dissolution [1, 9]: an identity question can be refused rather than answered, because what matters can survive without any verdict on identity — §1's refusal to adjudicate forks is that move re-run on a ledger. What we do *not* borrow is his positive claim. Parfit insists that fission does not divide what matters: "the relation of the original person to each of the resulting people contains all that interests us — all that matters — in any ordinary case of survival" [9]. Theorem 1 makes the inherited prior scarce, conserved and divisible precisely so that it *must* split eight ways among eight forks. A reputation ledger is exactly the kind of thing Parfit argues psychological continuity is not, and that is why a ledger can be conserved where a stream of experience cannot.

Theorem 2a's logic is Akerlof's market for lemons [2], run inside one identity — and it is one end of a continuum Grossman drew in 1981. His §2 shows that when quality is verifiable at negligible cost a monopolist *voluntarily* discloses in full, because a buyer with rational expectations reads any withheld information as unfavourable [10]; Milgrom proves the same conclusion game-theoretically — "at every sequential equilibrium of the sales encounter game, the salesman uses a strategy of full disclosure" [11]. His §3 takes the opposite extreme, disclosure costlier than the whole quality spread, and lands on our Theorem 2a: "we will have the usual 'lemons' problem ... each high-quality seller will want to be distinguished from those of average quality, but in this case there is no way for him to do so." So Grossman's §2/§3 pair *is* our Theorem 2b/2a pair, indexed by a disclosure-cost parameter we do not model — and we have accordingly renamed Theorem 2a, since in this literature "the unraveling result" names the §2 conclusion, which is full disclosure, not collapse.

That renaming exposes a real question about Theorem 2b, and answering it is the sharper contribution. If unraveling holds, the daemon need only *permit* attestation and voluntary disclosure will do the rest; our mandatory-attestation premise would be stronger than the economics requires. The unraveling result needs several strong hypotheses [12], and at least two fail here. Attestation is not costless. And more decisively, buyers cannot detect concealment — Milgrom is explicit that his result assumes the buyer "can verify ... statements like: 'I have reported everything I know'," and in an agent marketplace a buyer cannot distinguish an agent that *declines* to attest from one whose runtime *cannot*. With that hypothesis gone, voluntary disclosure no longer unravels, which is why the daemon must furnish the attestation channel rather than merely allow it. What Theorem 2b then adds to disclosure theory is that attestation does not only reveal the engine, it *re-prices* it.

A third hypothesis fails for a different reason: even a competitive *market* for attestation need not unravel to full disclosure. Lizzeri shows a monopoly certification intermediary optimally withholds everything and extracts the entire surplus whenever its own profit, not the buyer's welfare, sets the disclosure rule, and that full revelation is recovered only in the frictionless limit of infinitely many free-entering competitors — a limit any positive, sequential entry cost defeats, sustaining the pooling monopoly indefinitely [13]. A voluntary, third-party attestation market therefore inherits a failure mode a protocol-mandated channel does not: the daemon is not a profit-maximizing intermediary choosing how much of the engine id to reveal, so its incentive to pool never arises in the first place.

The observation that costly instruments only certain types will pay can *sort* them traces to Spence [3]; the deterrence instrument itself — post a stake forfeited on misconduct, sized to the gain — is Becker's expected-penalty condition [15], which is also what Theorem 2a's $qB \geq \Delta c$ restates. That zero-cost pseudonyms make sanctions void, and that newcomers must consequently pay dues, is Friedman and Resnick [4], with the replication half of the same coin as Douceur's Sybil attack [7]; Feldman et al. [14] quantify the resulting social cost against the turnover rate, finding it bites only when turnover is high — Theorem 4 takes that penalty's *level* as given by deterrence and asks the question they do not, which is what *shape* it should have over time. Reputations as tradeable assets attached to names is Tadelis [5]; their value and decay under imperfect monitoring is Mailath–Samuelson [6]. None of these components is ours.

One citation is owed and unresolved. Theorem 4's foil is the deferred-compensation instinct [8]; we have not been able to obtain Lazear's article, and if his contract front-loads the honest worker's implicit bond then Theorem 4 agrees with him rather than correcting him. We state the foil as folklore until that is checked.

New, honestly. The contribution is the assembly for software agents, where forking, distillation, engine substitution, and resurrection are cheap, routine, first-class operations rather than edge cases: (1) the transfer-semantics conservation theorem for reputation inheritance, with the refuted budget-only phrasing reported and the 8.2× copy-fork exhibit of what it pre-

vents; (2) the observation that engine attestation does not merely deter substitution but, under a stated pricing hypothesis, *flips* its incentive to the planner's efficiency rule at zero audit stake — with the Bertrand case, where the flip collapses, named — plus the scope lemma — price per (principal, engine), sanction per principal — that a machine-checked migration protocol shows is critical in both directions; (3) resurrection soundness as a theorem with three named attacks, each caught by a shortest trace when its clause is dropped; and (4) the probation-cliff optimality result, which sharpens "newcomers pay dues" into "the dues schedule has a unique optimal shape, front-loaded to the newcomer's ceiling rather than ramped — and a feasibility bound past which probation is the wrong instrument entirely." Where a claim rests on a bounded model or a falsification sweep rather than a general proof, the section's boundary block says so at the same prominence as the claim.

Reproducibility

Every [internal] number regenerates from three self-contained Python scripts at seed 20260816, shipped alongside this paper: `a6_no_mint.py` (Theorem 1: the closure-sum refutation, the 4000-DAG sweep, the copy-full mutant's 2-op mint and 8.2× quorum multiplication), `b5_engine_substitution.py` (Theorems 2–3: the 4000×8 price-independence sweep, threshold and stake lines against simulation, 2000 death spirals, the attested IC-sign check, the 747-state migration machine, and all six mutants with shortest traces), and `b6_probation.py` (Theorem 4: 76,000 deterrence-tight schedules against the cliff, plus the closed-form exchange spot check — all under $g_t \geq 0$ with no per-period cap, so this script covers Theorem 4(ii) and not the capped clause 4(iii); a $g_t \leq L$ clip is the open script task). Figures 4 and 5 regenerate deterministically from `figures/src/paper5_figures.py`; Figures 1 and 3 from `figures/src/r12_figures.py`; and every balance in the lineage DAG of Figure 2 is the ledger in `a6_no_mint.py` replayed operation by operation. Every [verified] number is recomputable from the closed forms stated in the boxes with a hand calculator.

References

[1] D. Parfit. *Reasons and Persons*. Oxford University Press, 1984.

[2] G. A. Akerlof. The market for "lemons": quality uncertainty and the market mechanism. *Quarterly Journal of Economics*, 84(3):488–500, 1970.

[3] M. Spence. Job market signaling. *Quarterly Journal of Economics*, 87(3):355–374, 1973.

[4] E. Friedman and P. Resnick. The social cost of cheap pseudonyms. *Journal of Economics & Management Strategy*, 10(2):173–199, 2001.

[5] S. Tadelis. What's in a name? Reputation as a tradeable asset. *American Economic Review*, 89(3):548–563, 1999.

[6] G. J. Mailath and L. Samuelson. Who wants a good reputation? *Review of Economic Studies*, 68(2):415–441, 2001.

[7] J. R. Douceur. The Sybil attack. In *Proc. 1st International Workshop on Peer-to-Peer Systems (IPTPS)*, LNCS 2429, pp. 251–260, 2002.

[8] E. P. Lazear. Why is there mandatory retirement? *Journal of Political Economy*, 87(6):1261–1284, 1979.

[9] D. Parfit. Personal identity. *The Philosophical Review*, 80(1):3–27, 1971.

[10] S. J. Grossman. The informational role of warranties and private disclosure about product quality. *Journal of Law and Economics*, 24(3):461–483, 1981.

[11] P. R. Milgrom. Good news and bad news: representation theorems and applications. *Bell Journal of Economics*, 12(2):380–391, 1981.

[12] D. Dranove and G. Z. Jin. Quality disclosure and certification: theory and practice. *Journal of Economic Literature*, 48(4):935–963, 2010.

[13] A. Lizzeri. Information revelation and certification intermediaries. *RAND Journal of Economics*, 30(3):214–231, 1999.

[14] M. Feldman, C. Papadimitriou, J. Chuang, and I. Stoica. Free-riding and whitewashing in peer-to-peer systems. In *Proc. ACM SIGCOMM Workshop on Practice and Theory of Incentives in Networked Systems (PINS)*, pages 228–236, 2004.

[15] G. S. Becker. Crime and punishment: an economic approach. *Journal of Political Economy*, 76(2):169–217, 1968.

[16] W. Cai, T. Shi, X. Zhao, and D. Song. Are you getting what you pay for? Auditing model substitution in LLM APIs. arXiv:2504.04715, 2025.