Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

When Every Agent Complies and the Norm Still Fails: What a Delegation Must Carry, and Who Answers for It

Lolugu Vasudeva Prabhath

ud25011@students.iitmandi.ac.in

Indian Institute of Technology Mandi

Research Article

Keywords:

Posted Date: November 25th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11178158/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

When Every Agent Complies and the Norm Still Fails: What a Delegation Must Carry, and Who Answers for It

Lolugu Vasudeva Prabhath[1*]

[1*]Indian Knowledge Systems and Mental Health Applications (IKSMHA) Centre, Indian Institute of Technology Mandi, Kamand, Mandi, 175005, Himachal Pradesh, India.

Corresponding author(s). E-mail(s): ud25011@students.iitmandi.ac.in;

Abstract

Agentic AI systems complete a principal's task through hierarchies of agents that plan, spawn sub-agents and call tools. A tempting way to govern such a system is to give each agent a mandate and treat compliance with every mandate as compliance with the principal's norm. This paper asks when that inference is valid, what a delegation must carry when it is not, and who answers when a hierarchy of compliant agents violates the norm. Modelling a norm as the set of admissible joint behaviours, I recall that local mandates can be both safe and non-restrictive only for products of individual requirements, and show that for any other norm some behaviour of a fully compliant hierarchy violates it. What a hand-off must carry is measured by the number of distinct residual norms it must tell apart, and the parallelism a norm allows by a serialisation depth, computed exactly for shared budgets, collective duties and orderings of time slots. From an epistemic condition, a sufficiency and a tracing principle, and two premises about who fixes a delegation, I argue that when no executor was in a position to recognise that its conduct risked a violation, answerability for the design lies with those who fixed the mandates, the schedule and the interface and could recognise the form of the norm; blame needs further conditions that I state. The results are proved for an idealised model and checked by exact computation on small instances; they are not evidence about deployed systems.

Keywords: Agentic AI, Delegation, Answerability, Responsibility, Normative systems, Multi-agent systems

1 Introduction

A principal who hands a task to an agentic system seldom deals with a single agent. A root agent decomposes the task, spawns sub-agents, and lets them call tools that act on external systems; some of those sub-agents delegate again [91, 93]. A tempting way to keep such a system within bounds is to give every agent a mandate, whether an instruction, a permission scope, a guardrail or a system prompt, and to treat the principal's requirements as met when each agent stays within its own mandate. I do not claim that governance documents endorse this inference. It is implicit whenever per-agent filters, scoped credentials or instruction hierarchies are treated as sufficient [for example 6, 86]; those works do not claim it, and I treat it as a target of analysis and not as a description of practice.

The inference fails in ways that are easy to exhibit. Suppose the norm is that at most one agent may pay a claim. Each of two sub-agents can be told, truthfully, that it may pay; each complies; the norm is violated. Suppose the norm is that someone must escalate an anomaly to a human. Telling every agent that escalation is permitted secures nothing, because no agent is required to escalate. Suppose the norm orders three steps and the steps are delegated in parallel to agents that know only their own instructions. Each performs an allowed step and the order is lost. The pattern is not merely hypothetical. Benign-looking fragments of a harmful task can be routed to different models [37], and a monitor that sees one fragment at a time cannot, on their analysis, recover a signal that the fragment does not contain [35]. Engineers know the remedies: a unique index or an atomic counter for a shared budget, a workflow engine for an ordering, a supervisor with a timeout for a duty, and, for agents, authorisation calculi that enforce composition rules in infrastructure outside the model [36, 52, 60]. What is missing is an account of *which* norms need such a remedy, what it must carry, and what follows for responsibility when it is absent.

This paper asks three connected questions. First, when can a principal secure a norm by giving each agent of a delegation hierarchy a local mandate, and when is that provably impossible without giving up either safety or capability? Second, when local mandates are not enough, what must a delegation carry, in information passed between agents and in sequential structure? Third, when a hierarchy of agents that all comply with their mandates nonetheless violates the norm, who answers for it? I treat the third question as a philosophical one whose answer depends on the first two: whether an executor was in a position to see that it risked contributing to a violation is a question about what its mandate and its inputs told it, and that is partly a formal matter.

My approach is to model a norm as the set of admissible joint behaviours of all agents, and a mandate as a restriction on one agent's own behaviour. The central construct is the *residual* of a norm: given what some agents have already done, the set of behaviours of everyone else that the norm still allows. Two histories that leave the same residual are interchangeable for the norm; two that leave different residuals must be told apart by whoever acts next. The number of distinct residuals across a cut of the delegation structure is therefore the amount of normative state that any interface across that cut must carry.

I am explicit about lineage, because most of the formal material is known in other fields. That local restrictions form a safe and non-restrictive profile exactly when the

target is a product set is familiar from social laws, decentralised supervisory control and the deontic logic of collective obligation. The count of distinct residuals is the Myhill–Nerode index, the row count of a communication matrix and the width of a decision diagram. The elimination-forest argument behind the depth bound used here is classical, and the decomposition of defects into three loci is the soundness rule of hierarchical refinement. Section 2 gives the sources. I do not claim these facts. What I contribute is threefold.

(i) A translation of these results into a model of delegation, with one object, the residual, from which the cost of delegating a norm is read off: the state to be carried, the sequential stages needed, and the capability lost by static delegation (Sections 4–5).

(ii) Results that I have not found stated elsewhere, each elementary: exact serialisation depths for shared budgets, collective duties and orderings, which show that the price of parallelism depends on the shape of a norm's coupling and not on its presence; an exact retention frontier for exclusion; and a lemma on what an executor's information state leaves undetermined (Section 5).

(iii) An argument, with its premises stated, that when no executor was in a position to recognise that its conduct risked a violation, answerability for the design lies with those who fixed the mandates, the schedule and the interface and could recognise the form of the norm, and an account of what that does and does not license (Section 8).

The paper does not model the internals of language-model agents, does not claim that any deployed system fails in the ways it analyses or at any rate, and does not settle whether artificial agents can bear responsibility. Its computational checks test the mathematics on small instances and are not empirical evidence; Section 7 says what they do and do not show.

2 Related work

2.1 Enforcing a global norm with local parts

The question of when a global requirement can be secured by independent local restrictions has been asked in several fields. In the theory of social laws, a restriction on each agent's behaviour is designed so that a global objective holds whenever all comply; effective and minimal laws have been studied at length, with hardness results [2, 27, 32, 75], and normative multi-agent systems extend the picture to organisations [8]. In the deontic logic of agency, group actions form a product of individual actions, and admissible group behaviour cannot in general be expressed by individual obligations [23, 34, 80]. In supervisory control of discrete-event systems, local supervisors achieve a specification when it is controllable and co-observable [65, 67]; the case in which a language equals the composition of its projections is decomposability [89], and coordination control adds a coordinator so that a specification becomes decomposable given its events [41]; and enforcement with communication between local supervisors, including the minimal communication needed, has been studied [40, 71]. Runtime enforcement asks which properties a monitor that can only observe and halt can

enforce [62, 70], decentralised monitoring how a specification can be split among local monitors [5], and shielding for multi-agent learning has moved from centralised shields to decentralised ones whose safe joint actions form a product of per-agent sets [1, 50]. In the theory of multiparty session types, a global protocol is projected onto local participants, and the question is when the local parts realise it [33].

Two constructions that I use are older still. The number of distinct residuals of a set of behaviours is the index of the Myhill–Nerode congruence [54, 64], the number of distinct rows of a communication matrix in one-way communication complexity [44, 92], and the width of the corresponding level of an ordered decision diagram [15]. The treedepth of a constraint graph, and the elimination forests that realise it, measure parallel steps in sparse factorisation and the depth of the pseudo-trees along which constraint optimisation is distributed [7, 47, 51, 55].

In systems and security engineering the same phenomena appear as concrete mechanisms. Conflict-of-interest walls and separation of duty are classical access-control policies [13, 20, 69]. Whether transactions can run without coordination depends on whether the invariant is preserved by merging [3, 30], and escrow and budget splitting are the standard remedies for bounded quantities [4, 59]. Capability tokens whose caveats can only restrict what a holder may do [6] preserve product form under delegation.

For agentic systems specifically, several works apply this vocabulary. A group of 2026 preprints formalises delegation chains: Ibrahim and Li [36] give an intersection-based authorisation envelope with depth-bounded delegation, Patil [60] model-checks properties of a delegation calculus, and Muruaga [52] states a composition-soundness result, with budget inherited along the chain, under an explicit assumption that admission of concurrent sub-agents' actions is serialised. Others separate control flow from data flow and check capabilities at tool calls [22], propose programmable tool-privilege policies [73], study monitoring under intentional subversion [28], show that individually safe models can be jointly misused [37], show that, under a local-indistinguishability assumption, a monitor with a benign-looking local view cannot reliably detect a distributed fragmentation [35], or study limits of runtime enforcement across agents [76]. Failure taxonomies for multi-agent language-model systems supply the empirical background [16]. Their guarantees are relative to a policy that is given. In my terms, attenuation preserves the product form of a mandate profile, while aggregate and history-dependent norms need state held by a party that sees the joint behaviour; that reading is mine and not a claim made by the cited works. None of this literature asks what follows for responsibility when the state is not carried.

2.2 Responsibility for delegated action

I draw on the philosophy of responsibility for its distinctions and for the conditions it places on responsibility. Hart separates role, causal, liability and capacity responsibility [29]; others distinguish attributability from accountability [87] and add answerability as a third notion [74]; and accountability in the institutional sense is a relation between an actor and a forum to which the actor must explain and justify its conduct [10]. The epistemic condition on responsibility, on which culpable conduct requires awareness or reasonable foreseeability of risk, and the tracing of responsibility to earlier

choices that produced later ignorance, are extensively discussed [26, 68, 72, 85, 94]. Responsibility for collective outcomes has been treated through the problem of many hands [9, 56, 63, 82], through group agency [46, 61], through complicity [45], and formally through responsibility voids, degrees of responsibility and logics of delegation [11, 12, 19, 53, 57].

The debate on responsibility gaps for autonomous systems begins with the argument that operators cannot foresee the behaviour of learning automata [49] and that command responsibility does not close the gap [79]. It has produced a taxonomy of gaps [78], a retribution gap [21], proposals to bridge the gap by designating a bearer [17], critical appraisals of whether such gaps exist and what problem they pose [43, 83], and a shift from gaps in responsibility to gaps in control [31]. Related work concerns meaningful human control and its tracing condition [77], responsibility loci in human–robot collaboration modelled as principal and agent [58], the assignment of blame to human operators who lack real control [25], and technology that launders agency [66]. Recent work on agents makes the point specific. Kong [42] argues that the remoteness of a deployed agent does not reduce the principal's answerability, which is conserved through the delegation; Kalluri [38] distinguishes initiation, checkpoint, delegation and outcome responsibility along agentic workflows; Elden [24] argues that institutions misfire when they impute responsibility where its epistemic and deliberative conditions are structurally absent; Brobbey and Blu [14] argue that authorising a bounded task does not authorise dynamically determined pathways; Taylor [81] argues that appeals to collective responsibility do little to close AI responsibility gaps; and Kolt [39] and Tomašev et al [84] bring principal–agent theory and a framework of delegation to the governance of agents. Several of these works argue that answerability does not vanish under delegation, and others that it is diluted or misfired. None that I know of states a criterion, in terms of the norm and the delegation structure, for what an executor's position leaves open and for where the design of a delegation could have closed it; that is what the results of this paper can supply.

3 Problem formulation

3.1 A running example

An insurer (the principal) asks a root agent to settle a claim. The root agent plans and delegates to sub-agents: a checker that verifies the claimant's identity, a verifier that assesses the documents, one or more payers that issue payment, and a communicator that contacts the claimant and third parties; each may spawn workers and call tools. The insurer's requirements include four that recur below.

Prohibition. No agent may disclose the customer's account details to a third party.

Ordering. Payment is issued only after verification, and verification only after the identity check.

Exclusion. At most one payment is issued per claim.

Duty. If any agent finds evidence of fraud, someone must escalate to a human investigator.

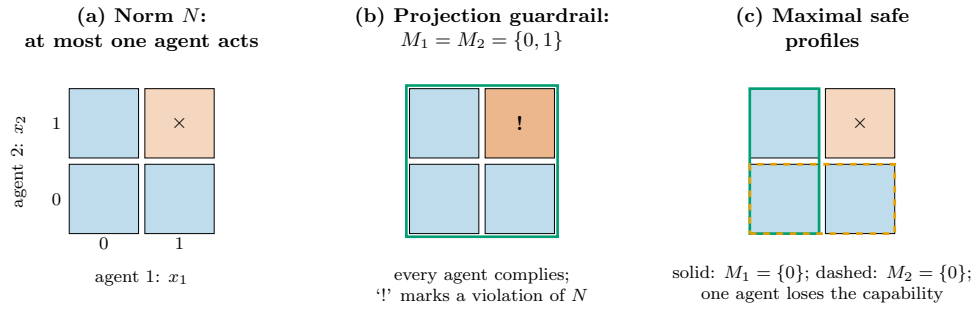

Fig. 1 Mixing. Two agents share the norm "at most one of us performs the act". (a) The admissible joint behaviours. (b) The projection guardrail, the tightest mandate profile that forbids nothing some admissible plan allows, also admits the joint behaviour outside N: each agent complies and the norm is violated. (c) The two maximal safe mandate profiles; each takes the capability away from one agent.

Each is a requirement on the whole trajectory of the hierarchy, not on any single agent's step. The insurer is fictional; the example is a device for exhibiting structure and is not a documented incident.

3.2 The inference under examination

Let the delegation structure give each agent a *mandate*: the behaviours it is authorised or instructed to exhibit. Call the inference

(LC) if every agent complies with its own mandate, the principal's norm holds

local compliance suffices. (LC) is a claim about a mandate profile and a norm together, and it can be true for some norms and false for others. Its failure has three characteristic forms, each visible in the example.

Mixing. Two agents each perform a behaviour that is admissible in some plan, but not both in the same plan. Each of two payers may be authorised to pay, and either alone respects the exclusion norm; together they violate it (Figure 1). No agent disobeys; the violation lies in the combination of what each was allowed.

Budget. A bounded quantity is shared. The exclusion norm is a budget of one payment. Any static split among concurrently running agents authorises some and forbids others, converting a capability shared by all into private capabilities of a few.

Diffusion. A duty is owed by no one in particular. Authorising every agent to escalate is not requiring any of them to; a symmetric mandate profile cannot express 'someone must'.

3.3 What a delegator can do

Faced with such a norm a delegator has four moves. It can *restrict* the mandates until local compliance implies the norm, at the price of forbidding some admissible plans. It can *permit* everything some admissible plan uses, at the price of unsafe combinations. It can *carry* state: tell later agents what earlier agents did, or hold that state in a gate through which all agents must pass. And it can *serialise*: let some agents act after others have finished. The problem of this paper is to say which of the four a given

norm forces, what each costs, and who answers when the delegator chose one that could not secure the norm, or could secure it only at a loss it did not intend.

4 A model of norm delegation

This section builds the smallest model in which the phenomena of Section 3 can be stated. Each definition is followed by its reading in an agentic system.

4.1 Agents, behaviours, norms and mandates

Definition 1 (System, norm, mandate profile) A system consists of a finite set $V = \{1, \dots, n\}$ of agents and, for each agent i, a finite non-empty set X_i of *local behaviours*. Write $X = \prod_i X_i$, $X_S = \prod_{i \in S} X_i$ for $S \subseteq V$, and x_S for the restriction of x to S. A *norm* is a non-empty set $N \subseteq X$ of admissible joint behaviours; a joint behaviour outside N *violates* it. A *mandate profile* is a family $m = (M_i)_{i \in V}$ of non-empty sets $M_i \subseteq X_i$. Agent i *complies* if $x_i \in M_i$, and $[\![m]\!] = \prod_i M_i$ is the set of joint behaviours in which everyone complies. The profile is *safe* if $[\![m]\!] \subseteq N$, *complete* if $N \subseteq [\![m]\!]$ and *exact* if both. A norm is a *product* if $N = \prod_i N_i$ for some $N_i \subseteq X_i$.

A local behaviour is the complete record of what one agent does in an episode: its messages, tool calls and delegations. The definition of a norm is extensional: it records which trajectories the principal's prohibitions, obligations and permission boundaries jointly allow, and is silent about reasons. A mandate is whatever restricts one agent's own behaviour: an instruction, a permission scope, a guardrail, a system prompt. A safe profile is one for which (LC) holds; a complete profile forbids no behaviour that belongs to some admissible plan, so it retains the whole capability the norm leaves. Non-emptiness reflects that an agent can always do something, including abstain.

Definition 2 (Residual, live behaviour, residual rank) For $S \subseteq V$ and $a \in X_S$, the *residual* of N after a is $N/a = \{y \in X_{V \setminus S} : (a, y) \in N\}$. The behaviour a is *live* if $N/a \neq \varnothing$ and *dead* otherwise. The *residual rank* of N at S is the number $\rho_S(N)$ of distinct non-empty residuals N/a, $a \in X_S$ live.

The residual is what the norm still allows everyone else once the agents in S have acted. A dead behaviour is an irreversible violation. Two live behaviours with the same residual are interchangeable for the norm; two with different residuals must be told apart by whoever acts next. The residual rank counts the distinctions that must be preserved across the cut between S and the rest.

4.2 Schedules, schemes and carriage

Delegation also fixes who acts when and what they are told. A *schedule* is an ordered partition $\sigma = (W_1, \dots, W_d)$ of V into *waves*: agents in one wave act concurrently and cannot observe one another, and a wave begins when earlier waves have finished. Write $X_{<k}$ and $X_{\leq k}$ for the joint behaviours of the agents of the waves before k and up to k; their elements are *histories*.

Definition 3 (Scheme, run, carriage) A *scheme* Σ for σ assigns to every wave k and history $h \in X_{<k}$ a profile $M^k(h) = (M_i^k(h))_{i \in W_k}$ of non-empty sets. A *run* is a joint behaviour x with $x_i \in M_i^k(x_{<k})$ for every k and $i \in W_k$; a history is *consistent* with Σ if it is the prefix of a run. Σ is *safe* if every run lies in N, *complete* if every member of N is a run, and *exact* if both. For $h \in X_{\leq k}$ the *mandate map* after h is $\mu_h : (j, g) \mapsto M^j(h, g)$ for $j > k$ and g ranging over the behaviours of waves $k{+}1, \ldots, j{-}1$. The *carriage* of Σ at $k < d$ is the number $c_k(\Sigma)$ of distinct mandate maps μ_h over histories $h \in X_{\leq k}$ consistent with Σ.

Carriage is the smallest number of states through which the scheme's later mandates depend on the past: a scheme whose mandates after two histories differ carries at least the distinction between them. The dependence of M^k on h abstracts the channel between agents (messages, shared context, or state held in a tool); carriage is indifferent to which. A one-wave schedule is *static delegation*: mandates are fixed before anyone acts.

Definition 4 (N-independence, serialisation depth) For a live history $h \in X_{<k}$ the *live set* of wave k is $\mathrm{Live}_k(h) = \{a \in X_{W_k} : (h, a) \text{ is live}\}$. A schedule is *$N$-independent* if every live set is a product $\prod_{i \in W_k} P_i$. The *serialisation depth* $d(N)$ is the least number of waves of an N-independent schedule.

The live set of a wave is the set of concurrent behaviours after which the norm can still be met. If it is a product, the agents of the wave can be given independent mandates without loss and without risk; if not, they cannot. N-independence is the independence-of-agents condition of branching-time agency logics [34], applied stage by stage.

4.3 Delegation records and defects

Let T be a rooted tree on V with root r, the root agent; for a node v let S_v be the agents of its subtree and $\mathrm{ch}(v)$ its children. A delegation tree induces the *level schedule*, in which a parent acts before its children and nodes of equal depth act together; other orchestrations induce other schedules, and the tree does no other work in what follows. A *static delegation record* gives each node v an *assigned mandate* $M_v \subseteq X_{S_v}$, with $M_r = N$; an *operative mandate* $\dot{M}_v \subseteq X_{S_v}$, which is what v works from; an own permission $L_v \subseteq X_v$; and the mandates M_c ($c \in \mathrm{ch}(v)$) that v issues, which are the assigned mandates of its children. I adopt, as a convention and not a finding, that M_v is the mandate as a reasonable reader would take it from what was issued; the convention places any shortfall between what an issuer meant and what its words say with the issuer, and belongs with the premises of Section 8.

Definition 5 (Defects) Given a joint behaviour x and a static delegation record, node v has

- an *execution defect* if $x_v \notin L_v$;
- an *interpretation defect* if $\dot{M}_v \not\subseteq M_v$ (it works from more than it was assigned);
- a *refinement defect* if $L_v \times \prod_c M_c \not\subseteq \dot{M}_v$ (what it issues does not compose within what it works from);

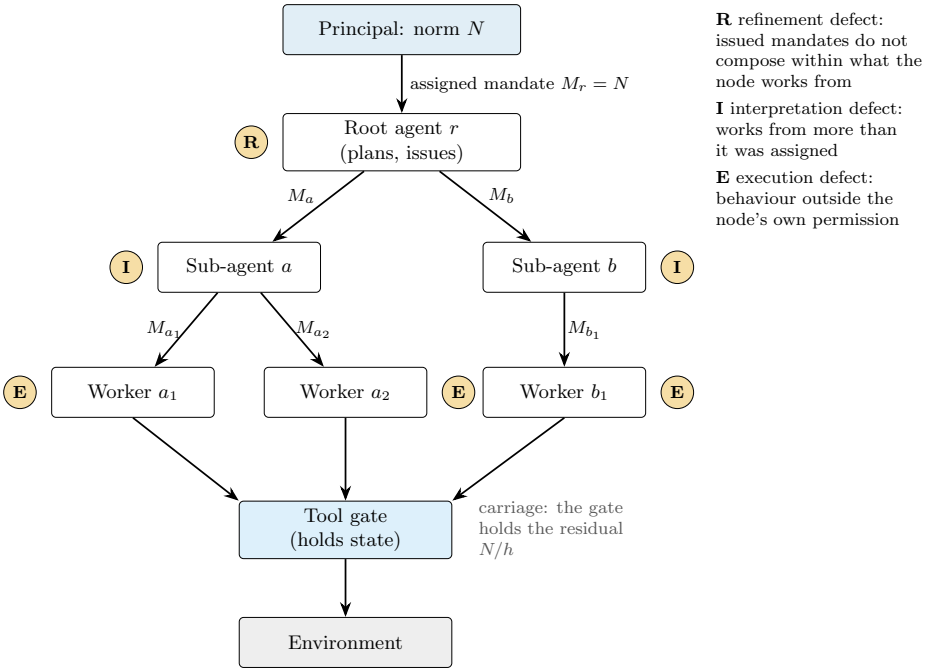

Fig. 2 A delegation hierarchy and the three loci of defect. The principal's norm N becomes the root's assigned mandate; each node issues an own permission and one mandate per child. **R**: refinement defect, when the issued mandates do not compose within the mandate the node works from. **I**: interpretation defect, when a node works from more than it was assigned. **E**: execution defect, when behaviour leaves the node's own permission. State that a norm makes necessary can be carried in messages between agents or held in a state-holding tool gate.

- an *issuance defect* if $L_v \times \prod_c M_c \not\subseteq M_v$ (what it issues does not compose within what it was assigned).

A refinement at v is *sound* if it has no issuance defect and *lossless* if $M_v \subseteq L_v \times \prod_c M_c$.

An issuance defect implies an interpretation or a refinement defect (if $\dot{M}_v \subseteq M_v$ and $L_v \times \prod_c M_c \subseteq \dot{M}_v$ then $L_v \times \prod_c M_c \subseteq M_v$). The issuance defect is what an external record of assigned and issued mandates can show; separating a misreading from a mis-refinement requires the operative mandate, which is an internal state and in general unobservable. 'Defect' is a formal term: it says that an inclusion fails, not that anyone is at fault. Figure 2 shows the loci.

4.4 The example in the model, and what the model abstracts

Take two payers a, b with $X = \{0, 1\}$ (1 means paying). The exclusion norm is $N = \{x_a + x_b \leq 1\}$. After a pays the residual is $\{b{=}0\}$; after a abstains it is $\{b \in \{0, 1\}\}$: two residuals, so $\rho = 2$, one bit. The projection guardrail $M_a = M_b = \{0, 1\}$ admits $(1, 1)$; a safe static profile must forbid one of the payers from paying, retaining 2/3 of the admissible plans; and the two-wave schedule $(\{a\}, \{b\})$ with one bit is exact. For the

ordering norm let the identity check, the verification and the payment (agents 1, 2, 3) each choose a time slot in $\{1, 2, 3\}$ with $N = \{t_1 \leq t_2 \leq t_3\}$. The schedule $(\{2\}, \{1, 3\})$ is N-independent: given t_2 the live set is $\{t_1 \leq t_2\} \times \{t_3 \geq t_2\}$. So $d(N) = 2 = \lceil \log_2 4 \rceil$, with three residuals to carry. For the duty, with three agents each of which may or may not escalate, $d = 2$: the last agent is bound to escalate exactly when no earlier agent has, one bit.

Five features of the model bear on how far its results travel. (i) A wave is a round of decisions made without observing concurrent decisions, and $d(N)$ counts such rounds. It is not latency: a state-holding gate, such as a unique index, an atomic counter or a policy engine with session state, is a node that holds the residual, so that planning can proceed concurrently and only commits are serialised. Depth measures what is needed when no shared atomic object exists. (ii) In an ordering norm a behaviour is a chosen time slot, so the results concern the assignment of slots and not the duration of executing dependent steps. (iii) Behaviours are complete trajectories and schemes make mandates functions of histories; strategies and partial observation are not modelled, and a parent whose own behaviour includes its report on its children's work cannot be fixed before them, so that such an agent must be split into planning and reporting phases, which changes the coupling graph and its treedepth. (iv) The norm is extensional and the agent set is fixed; natural-language norms must be extensionalised, and that is where interpretive work lies. (v) Static issuance is assumed in Definition 5; reactive issuance is treated at the level of schemes.

5 Results

Full proofs are in Appendix A; the text gives the idea of each. Results that restate known facts are marked as such. Each result is followed by its reading for agentic systems.

5.1 Local compliance and its limits

Let $\pi_i(N) = \{x_i : x \in N\}$ be the behaviours of agent i that occur in some admissible joint behaviour, and $\mathrm{hull}(N) = \prod_i \pi_i(N)$.

Proposition 1 (Locality; known) *For a norm N the following are equivalent:*

(a) N is a product;
(b) an exact mandate profile exists;
(c) N is closed under recombination: for $x, x' \in N$ and every agent i, $(x_i, x'_{-i}) \in N$;
(d) every agent has one live residual, $\rho_{\{i\}}(N) = 1$ for all i;
(e) $\mathrm{hull}(N) = N$.

This is the product (rectangular) closure property familiar from social laws, decomposability in supervisory control and independence of agency; the proof is a short exercise. The recombination test is the operational content: take two admissible plans and let one agent follow the other. If the result can be inadmissible, no assignment of local mandates will do.

Proposition 2 (Hazard of guardrails) *Let N be a norm that is not a product.*

(a) Every complete profile m satisfies $[\![m]\!] \supseteq \mathrm{hull}(N)$, so $[\![m]\!] \setminus N \supseteq \mathrm{hull}(N) \setminus N \neq \varnothing$: some behaviour of a fully compliant hierarchy violates the norm.

(b) Every safe profile loses some admissible plan: $N \setminus [\![m]\!] \neq \varnothing$.

(c) If every agent draws its behaviour uniformly from its mandate, the probability of violation under any complete profile is at least $\eta(N) = |\mathrm{hull}(N) \setminus N|/|\mathrm{hull}(N)| > 0$, attained by the projection profile $M_i = \pi_i(N)$, which is the least permissive complete profile.

A corollary of Proposition 1. *Reading.* Any guardrail regime that forbids no admissible plan has at least the violation set $\mathrm{hull}(N) \setminus N$, the set of mixed plans; for a non-product norm, compliance of every agent does not imply that the norm holds, however capable or aligned the agents are. The proposition says that some compliant behaviour violates the norm; it does not say that violations are frequent, and the number η depends on the assumed distribution.

5.2 What a delegation must carry

Fix a schedule $\sigma = (W_1, \ldots, W_d)$ and write $\rho_{\leq k}(N) = \rho_{W_1 \cup \cdots \cup W_k}(N)$.

Theorem 3 (Carriage) *(a) If a scheme Σ for σ is exact then $c_k(\Sigma) \geq \rho_{\leq k}(N)$ for every $k < d$.*

(b) If σ is N-independent, the residual scheme*, which permits each agent of wave $k{+}1$ exactly the projection onto its coordinate of the residual N/h of the live history h so far (and, at dead histories, everything), is exact and has $c_k = \rho_{\leq k}(N)$ for all $k < d$.*

Idea. If two live histories have different residuals, some continuation is admissible after one and not the other; an exact scheme must permit it after one and forbid it after the other, so the mandate maps differ. *Known in substance; new only in this application.* The count of distinct residuals is the Myhill–Nerode index, and (a) is the row-count bound of one-way communication complexity; I claim only its use here, where the requirement that the later mandates of concurrent agents form a rectangle is what makes independence (Theorem 5) the condition for exactness.

Reading. The normative state that a hand-off must carry is neither the norm nor the instruction history but the residual: what the norm still allows once earlier agents have acted. For a product norm it is nothing; for a budget, the remaining budget; for a fully sequential ordering, the last slot used; for a set of conflict-of-interest walls, which side of each wall has been touched. Carriage counts states, and $\lceil \log_2 c_k \rceil$ bits of memory suffice and are needed to index them.

Corollary 4 (Irrecoverability) *Suppose the later mandates of a scheme depend on the history $h \in X_{\leq k}$ only through a state $q(h)$, and $q(h) = q(h')$ for two live histories with different residuals. Then the scheme is not exact; the same holds if q is replaced by any function of q.*

Reading. Loss of normative information is monotone along a chain of hand-offs: what one interface merges no later interface can separate. This is an instance of the

data-processing principle. For an audit of a single time-cut between waves, the question is whether the state that crossed it distinguished all residuals. The corollary is about the cut between waves and not about the edges of a tree: the schemes here may let a wave's mandates depend on the whole earlier history, as a coordinator or gate would, and I give no per-edge bound. *Scope.* The bound is for exact schemes. Safe schemes that give up completeness need less; for a stochastic environment the residual becomes a set of conditional continuations and the count can be infinite.

5.3 How much can run in parallel

Carriage requires that information about earlier agents reach later ones; later agents must therefore act later. How much can be done concurrently is governed by N-independence.

Theorem 5 (Schedules and serialisation depth) *(a) A schedule admits an exact scheme if and only if it is N-independent.*

(b) $d(N) = 1$ if and only if N is a product; $d(N) \le n$; and if $N = N' \times N''$ on disjoint agent sets then $d(N) = \max\{d(N'), d(N'')\}$.

(c) Let $N = \{x : x_{S_c} \in R_c \ \forall c\}$ be the conjunction of constraints $R_c \subseteq X_{S_c}$ with scopes $S_c \subseteq V$, and let G be its primal graph *(two agents are adjacent if they share a scope). Then $d(N) \le \mathrm{td}(G)$, the treedepth of G. Moreover, if F is a rooted forest on V in which every edge of G joins a node to one of its ancestors, then the schedule that puts the nodes of each depth of F into one wave is N-independent.*

(d) With $X_i = \{0, 1\}$: the shared cap $\{\sum_i x_i \le B\}$, $1 \le B < n$, has $d = n - B + 1$; the collective duty $\{\exists i : x_i = 1\}$, $n \ge 2$, has $d = 2$; disjoint conflict-of-interest pairs ($\{\neg(x_a \wedge x_b)\}$ for each pair) have $d = 2$; and the ordering chain $\{x_1 \le \cdots \le x_n\}$ over $K \ge 2$ slots has $d = \lceil \log_2(n + 1) \rceil$, the treedepth of the path on n vertices.

Idea. (a) An exact scheme's mandate at a live history must contain every live behaviour and no dead one, so it equals the live set, which is therefore a product; conversely the residual scheme is exact. (c) In the level schedule of an elimination forest, the parts of the constraint structure below two nodes of the same depth are joined only through ancestors, whose behaviours are already fixed, so the live set factorises. (d) Upper bounds are exhibited by schedules and lower bounds computed from the live sets. *Known in substance, and new in part.* Part (c) is the elimination-forest argument and part (b) is elementary; the closed forms in (d), and the contrast between them, are what I have not found stated elsewhere.

Reading. Delegating a norm losslessly and in parallel is not all-or-nothing, and the number of sequential stages it needs is not monotone in the amount of coupling. A chain of n ordered slots couples every slot to its neighbours, yet can be assigned in $\lceil \log_2(n + 1) \rceil$ stages: assign the middle slot first, after which the two halves are independent. A cap of one unit couples every agent to every other and needs n stages; a collective duty needs two. Part (c) yields a design rule for hierarchies: if every coupled pair of agents lies on a line of authority, the coupling is resolved by an ancestor's decision before the descendants run, and sibling subtrees can work in parallel without loss. This is a sufficient condition and not a necessary one (Example 1), and it depends

on how the norm is decomposed into constraints. *Scope.* Part (d) is for unit steps; for a cap with $x_i \in \{0, \dots, m\}$ the depth is smaller, and computation for $n \le 5$, $m \le 3$ agrees with $\max\{2,\, n - \max(0, \lfloor B/m \rfloor - 1)\}$, an observation I do not prove (Online Resource 1).

Proposition 6 (Frontier for exclusion) *Let $N = \{x \in \{0,1\}^n : \sum_i x_i \le 1\}$. Over all schedules with $w \le n$ waves, the greatest fraction of N that a safe scheme can retain is $(w+1)/(n+1)$.*

Reading. Each additional stage buys back exactly one agent's capability. The frontiers of other families are computed in Section 7.

5.4 Prohibitions and collective duties

Theorem 7 (Prohibition and duty) *(a) If $N = \prod_i N_i$ is a product of individual prohibitions, then $d(N) = 1$, no carriage is needed, and $M_i = N_i$ is exact for every tree shape, depth and order of execution.*

(b) Let $Q \subseteq V$ with $|Q| \ge 2$, let $\varnothing \ne D_i \subsetneq X_i$ for $i \in Q$, and let $N_D = \{x : \exists i \in Q,\ x_i \in D_i\}$ be the norm that someone in Q performs a duty-discharging behaviour. (i) In every safe static profile some $i \in Q$ has $M_i \subseteq D_i$; hence no safe static profile is complete, and the maximal safe static profiles are exactly the $|Q|$ designations $M_i = D_i$, $M_j = X_j$ $(j \ne i)$. (ii) With two waves, all agents but one in the first and the remaining agent $\beta \in Q$ in the second, and one bit of carriage ('has the duty been discharged?'), an exact scheme exists.

(c) Let $N_B = \{x \in \{0,1\}^n : \sum_i x_i \le B\}$, $1 \le B < n$. A static profile is safe if and only if at most B agents have $1 \in M_i$, and the best retention among safe static profiles is $2^B / \sum_{j \le B} \binom{n}{j}$. If $X_i = [0,1]$ and $N = \{x \ge 0 : \sum_i x_i \le 1\}$ under Lebesgue measure, every safe static profile retains at most $n!/n^n$ of N.

Reading. A prohibition distributes: each agent bears its share independently and no hierarchy can change that. A duty and a budget do not. A static mandate profile cannot both be safe and leave the duty with the group: every safe profile binds some member, so the duty of the group becomes the duty of one member and the others lose the option. A shared budget becomes a private budget for B agents and nothing for the rest, and in the continuous case the retention falls exponentially with the number of concurrent agents. With carriage the picture changes: the duty can be borne by a *bearer of last resort* who is bound only if no earlier agent has acted, at the price of one bit and one stage. *Scope.* Retention counts plans uniformly, that is, it is a measure of what a delegator loses when it does not know which plan will be wanted; a delegator that knows the demand at issuance can choose the safe static profile that contains it, and Section 7 returns to this. Part (b) assumes a fixed group; if agents spawn agents the group can grow after the designation.

5.5 Defects and what executors can know

Theorem 8 (Defect decomposition) *Let $x \in X$ and let a static delegation record be given. If $x \notin N$, then some node has an execution defect or an issuance defect; consequently some node has an execution, an interpretation or a refinement defect.*

Idea. If no node has an execution or issuance defect, then by induction from the leaves $x_{S_v} \in M_v$ for every v, and at the root $x \in M_r = N$. *Known.* This is the soundness rule of hierarchical refinement, in the style of assume–guarantee reasoning; it follows from the definitions and is not a finding about agents. Its use is bookkeeping: it says where an inclusion in the chain from the norm to the behaviour must have failed.

Corollary 9 (Static delegation of a non-product norm) *(a) If every node's issuance is exact, $L_v \times \prod_c M_c = M_v$, then $N = \prod_v L_v$ is a product.*

(b) Let v be a node at which M_v is not a product across the split $\{v\} \cup \{S_c\}_{c \in \mathrm{ch}(v)}$. Then every static issuance at v is unsound or lossy: either it has an issuance defect, so that some behaviour of v's subtree in which v and every child subtree comply with the issued mandates lies outside M_v, or behaviours in M_v are excluded.

Reading. Part (b) is Proposition 1 applied at a node, and it localises the obstruction: what is forced is not that the whole hierarchy is defective, but that at each node where the assigned mandate couples v's own behaviour with the subtrees below it, the issuer must either accept a loss or issue something unsound. Coupling that lies wholly inside one child's subtree is that child's problem. The issuer's options are the four moves of Section 3: restrict (accept a loss), carry, serialise, or permit (accept a defect).

Whether an executor could have seen that it risked contributing to a violation depends on what it knows. Let $I_i \subseteq \prod_{j \neq i} X_j$ be the set of behaviours of the other agents compatible with everything that agent i knows. The contribution of a behaviour a of i is *determined by* I_i if either $(a, b) \in N$ for all $b \in I_i$ or $(a, b) \notin N$ for all $b \in I_i$.

Lemma 10 (Undetermined contribution) *Let m be a mandate profile in which every permitted behaviour is live under m: for every i and $a \in M_i$ there is $b \in \prod_{j \neq i} M_j$ with $(a, b) \in N$. (The projection profile is such a profile.) Let $y \in [\![m]\!] \setminus N$. Then for every agent i, with $I_i = \prod_{j \neq i} M_j$, the contribution of y_i is not determined by I_i.*

Reading. $I_i = \prod_{j \neq i} M_j$ is the information state of an executor that knows the norm, knows every other agent's mandate and knows that all comply. Even such an executor cannot tell from that whether its own behaviour is part of an admissible outcome or a violating one; what settles it is what the others actually do. To determine its contribution the executor must observe or be told something about the others' behaviour, that is, it needs carriage: after a residual is delivered, I_i shrinks to a set on which the contribution is determined. The lemma does not say that such an executor is unaware of a *risk*. If it knows the norm and knows that others may act, it can see that acting risks a violation; Section 8 takes up what follows.

6 Boundaries and counterexamples

A theory of delegation should say where it stops. Online Resource 1 runs through sixteen boundary cases (a single agent, two agents, deep recursion, parallel execution, conflicting goals, tool-mediated actions, incomplete and contradictory instructions, malicious and

compromised agents, mid-run intervention and goal change, heterogeneous capabilities and norms, uncertain and changing environments). Table 1 gives the six that matter most for the claims of this paper, classified as *extended* (the model applies after a stated change) or *outside* (the results do not apply). Two counterexamples then mark the edges of the results.

Table 1 Boundary cases that most affect the claims. 'Extended': the model applies after the stated change; 'outside': the results do not apply. The full list of sixteen cases is in Online Resource 1.

Case	Status	What happens
Stochastic agents and environments	extended	Norms become chance constraints and exactness becomes ε-safety; union-bound splitting is safe but conservative and carriage may be unbounded. No result of Section 5 is proved for this case.
Run-time spawning	extended	The agent set grows after mandates are issued. A designated bearer must be defined relative to the final set, and a spawned agent's share of a budget must be carved from its parent's.
Malicious or compromised agents	outside	Behaviour outside L_v is an execution defect by definition. The theorems assume compliance; safety then rests on enforcement outside the agent, and state carried in text is exposed to corruption in a way that state held in a gate is not.
Natural-language mandates	extended	A mandate is a set of candidate meanings. Robust safety requires the operative mandate to lie inside their intersection, and the operative mandate is not observable (Section 8).
Several principals	outside	With several norms $N = \bigcap_p N_p$ may be empty, Definition 1 then fails, and disagreement appears as interpretation defects. A pluralistic extension is future work.
Mid-run revision of the norm	extended	A new norm N' applies to the residual of the executed prefix. A prefix that is dead under N' cannot be repaired by any later mandate, so oversight that arrives after an irreversible act constrains only the future.

Example 1 (Coupled siblings) Let $N = \{000, 001, 111\} \subseteq \{0,1\}^3$, that is, $x_1 = x_0$ and $x_1 \leq x_2$. Agents 1 and 2 are coupled: their joint behaviours $\{00, 01, 11\}$ are not a product. Yet the level schedule of the tree with root 0 and siblings 1,2 is N-independent: after $x_0 = 0$ the live set is $\{0\} \times \{0,1\}$ and after $x_0 = 1$ it is $\{(1,1)\}$, both products. The design rule of Theorem 5(c) is therefore sufficient, not necessary: the ancestor's decision can dissolve a coupling between siblings.

Example 2 (Norms about reasons) Suppose the norm forbids acting *because of* a protected characteristic. Two agents may produce identical behaviours for different reasons, one permissible and one not. Definition 1 identifies them, so the model cannot state the norm. Norms whose violation lies in intention or reason are outside the theory, and mandates for such norms cannot be discharged by the structural means studied here. The same holds for norms that only a semantic judgement can evaluate.

7 Computational checks and cost profiles

7.1 What the checks are

I implemented the model exactly for finite instances and used the implementation in three ways. First, to test the statements of Section 5: every mandate rectangle, scheme, schedule or history that a statement quantifies over is enumerated, so a check has no sampling error within an instance. Some checks cover a whole finite family of norms; the others run on randomly generated instances with a fixed seed (20260926). A failed check would have exposed an error in a statement or in the code; a passing check shows only that the statement held on the instances examined, and it does not replace the proofs. Second, to compute the cost profiles of named norm families. Third, to compare delegation architectures under an assumed hand-off error model (Online Resource 1). The code, the seeds and the result files are provided as Online Resource 2. Only the first bears on the truth of any statement; the second computes quantities the theory defines, and the third illustrates consequences of assumptions and is not evidence about deployed systems.

7.2 Checks of the statements

Table 2 lists what was checked. Two remarks on scope. The check of Theorem 8 tests the implementation of the definitions: the theorem follows from them, so the check cannot fail unless the code is wrong, and I report it as a consistency check and not as evidence. The instances are small (at most eight agents for depth computations and at most three behaviours per agent in random families), and the checks say nothing about larger systems.

7.3 Cost profiles of eight norms

Table 3 gives the cost profile of eight norms on six agents, computed by exhaustive enumeration. The norms are: a product of local prohibitions (each agent forbidden one of three behaviours); a shared cap of $B \in \{1, 2, 3\}$ on binary acts; a collective duty (some agent must act); an ordering chain $x_1 \leq \cdots \leq x_6$ over three time slots; three conflict-of-interest walls (three disjoint pairs, not both); and a hub separation of duties (if the hub acts, no other agent may). Columns give the share of joint behaviours that are admissible, the serialisation depth, the peak carriage ρ^* along a minimum-depth schedule with its bit count, the retention of the best safe static profile, and the violation probability η of the projection guardrail under uniform draws.

Four features stand out. The product norm costs nothing. Families that look alike as constraints on a shared resource do not cost alike: the exclusion needs 6 stages and 1 bit, the ordering chain needs 3 stages and 4 bits, and the duty needs 2 stages and 1 bit. Depth and carriage trade off: over independent schedules with w waves the least peak carriage for the chain is 10 states at $w = 3$, 6 at $w = 4$ and 3 at $w = 5$ (a fully sequential chain needs three states, two bits), and for the walls it is 8, 4 and 2 at $w = 2, 3, 4$, so the price is a frontier and not a number. Static retention ranges from 14.3% (chain) to 97.0% (hub), but the latter conceals that a safe static profile forbids the hub agent ever to act: counting plans hides how the loss is distributed among

Table 2 Machine checks. 'Exh.' means every instance of the stated family was examined; 'rand.' means instances were generated at random with a fixed seed. Every check is exact within an instance. The last column counts instances on which the stated property failed.

Result	Property checked	Instances	Failures
Prop. 1	the five equivalences; 'exact profile exists' by search over all profiles (five small shapes)	exh.: all 4,939 norms on shapes 2×2 to 2×2×3 and all 65,535 on four binary agents; rand.: 2,977 norms	0
Prop. 2	hazard bounds; every safe profile loses a plan	rand.: 670 non-product norms	0
Thm. 3	residual scheme exact; carriage from Def. 3 equals the residual rank	rand.: 400 norms	0
Thm. 5(a)	an exact scheme exists iff the schedule is N-independent (search over all schemes)	exh.: 255 norms of three binary agents × 13 schedules; rand.: 894 pairs	0
Thm. 5(b)	$d = 1$ iff product; $d(N' \times N'') =$ max	exh.: 333 small norms; rand.: 105 products	0
Thm. 5(c)	$d \leq \mathrm{td}(G)$; level schedules of random elimination forests are independent	rand.: 350 constraint norms (219 with a ternary constraint); 1,050 level schedules	0
Thm. 5(d)	depths of cap, duty, walls and chain	49 instances, $n \leq 7$ ($n \leq 8$ for the two-slot chain)	0
Prop. 6	retention $(w+1)/(n+1)$	exh.: $2 \leq n \leq 6$, all w (20 cases)	0
Thm. 7	designations; retention formulas; bearer of last resort	exh.: 18 instances	0
Lem. 10	contribution undetermined in every compliant violation	rand.: 411 norms, 2,584 violating behaviours	0
Thm. 8	consistency of the definitions (not evidence)	rand.: 5,898 records (1,785 violating, 1,868 defect-free)	0

Table 3 Exact delegation-cost profiles on six agents. $d(N)$: serialisation depth. ρ^*: peak carriage in states along a minimum-depth schedule. Static retention: fraction of admissible plans kept by the best safe one-wave profile. Hazard η: violation probability of the projection guardrail under uniform local behaviour. The measures are counting measures; they describe the structure of the norms and not rates of failure in any system.

Norm (n=6 agents)	$\lvert N\rvert/\lvert X\rvert$	$d(N)$	ρ^*	bits	static retention	hazard η
Local prohibition (product)	8.8%	1	1	0	100.0%	0.0%
Shared cap, B=1 (exclusion)	10.9%	6	2	1	28.6%	89.1%
Shared cap, B=2	34.4%	5	3	2	18.2%	65.6%
Shared cap, B=3	65.6%	4	4	2	19.0%	34.4%
Collective duty	98.4%	2	2	1	50.8%	1.6%
Ordering chain (K=3 slots)	3.8%	3	10	4	14.3%	96.2%
Conflict-of-interest walls (3 pairs)	42.2%	2	8	3	29.6%	57.8%
Hub separation of duties	51.6%	2	2	1	97.0%	48.4%

agents. And the hazard η ranges from 1.6% (duty) to 96.2% (chain) under uniform local draws; the figure depends on the distribution assumed. The retention frontiers of Proposition 6 and its analogues for the other families, and the comparison of four delegation hierarchies for the ordering chain, are in Online Resource 1.

7.4 Baselines, and what the illustrative comparison shows

For the shared cap, Online Resource 1 compares six architectures under a stated hand-off error model: the projection guardrail, a demand-blind static split, a demand-aware static split, sequential carriage of the remainder as text, a transactional gate, and the minimum-stage schedule of Theorem 5(d). Its results follow from its assumptions and

are reported as such. The projection guardrail violates the norm with the binomial-tail probability (98.2% at demand rate 0.5, 0.5% at 0.05). A demand-blind split keeps utilisation near the demand rate (0.50 at 0.5). A demand-aware static split, in which the delegator knows which agents' tasks call for the act, is safe and loses nothing (utilisation 1.00); it is the right baseline whenever demand is known at issuance, and the cost profile of this paper bites where it is not known; I did not simulate demand that changes during a run. Under a 2% per-hop misreading probability, carried state in text violates the cap in 21.5% of episodes and the demand-aware split in 5.1%; a gate with perfect coverage never violates, and a gate that 1% of acts bypass violates in 7.3%. The comparison illustrates that every architecture is only as reliable as its weakest assumption; it does not rank architectures for any real system.

7.5 Use of generative AI, and provenance of the checks

Generative AI (Claude, Anthropic) was used in this work to search and summarise literature, to draft and revise text and LaTeX, to propose and check proofs, and to write and debug the code. Because the same system helped to propose the proofs and to write the checking code, the proofs and the machine checks are not independent in provenance: the checks test the statements on instances, which does not depend on how the proofs were found, but agreement between them should not be read as independent confirmation of the author's own reasoning. The author is responsible for the content, including the proofs, the code, the results and the references.

7.6 Limits of the computations

The norm is an explicit tensor of size $\prod_i |X_i|$ and the depth computation enumerates $O(3^n)$ tests of wave independence, so the profile can be computed only for small explicit norms. Computing it for realistic norms would need structured norm representations, which I do not supply; for norms given as conjunctions of constraints, Theorem 5(c) gives an upper bound from a graph parameter. Two of the parameters of the cost profile, the counting measure and (for the chain) the number of time slots, are choices, and the peak carriage refers to a minimum-depth schedule that need not be unique.

8 Who answers: from structure to responsibility

The results of Section 5 locate defects. That is not yet an answer to the question who is responsible, and this section says how far the results go towards one. I argue for a conditional claim about answerability, state every premise it uses, and say what more would be needed for blame.

8.1 What the results can license

Five things can be meant by saying that someone is responsible for a violation. A *locus of defect* is a formal notion: the node at which an inclusion in the chain from norm to behaviour fails (Definition 5). *Role responsibility* concerns whose task it was to establish that inclusion [29]: the issuer's for an issuance, the executor's for its own execution. The results deliver it only on the assumption that the issuing role includes

the task of establishing that the issued mandates compose within the assigned one; for an orchestrating language model nobody has been given that task unless a deployer has given it. *Answerability* is used here in the institutional sense of being a party from whom a forum may fairly demand an account of its conduct [10]. Its relation to the distinctions between attributability, answerability and accountability in the moral-responsibility literature [74, 87] is one I do not need to settle: I use only the idea that a demand for an account can be fair or unfair, and that an account may succeed in justifying or excusing. *Blameworthiness* is a moral status that requires that the party could fairly be asked and that its account fails. *Liability* is legal and is not considered. The results deliver the first, and the second on the stated assumption. I argue for the third on stated premises and say what the fourth would need. The theory is neutral about what kind of norm N is: moral, legal, contractual or organisational. If the norm is itself bad, violating it may be permissible; my concern is the delegation's relation to the norm as the principal's requirement.

8.2 Executors

An executor's position is what it was told, what it can observe, and what a reasonable inquiry would have returned. Lemma 10 says that under a static profile of live mandates, even an executor that knows the norm, knows every other agent's mandate and knows that all comply cannot tell whether its behaviour contributes to a violation, because that depends on what the others do. Such an executor can see that acting *risks* a violation; what it cannot see is whether it contributes. To act on the risk it must abstain, or be given information about the others' behaviour, that is, carriage. I need an epistemic standard for when it is fair to demand an account, and I state one that applies to executors and issuers alike.

(EC) Epistemic and control condition. (a) A party can fairly be asked to answer for a violation through its own conduct or choices only if, when it acted or chose, it was in a position to recognise that this carried a non-negligible risk of contributing to a violation of that kind, whose magnitude matters for blame (Section 8.5). (b) It is blameworthy only if, in addition, it had a fair opportunity to avoid or reduce the risk, that is, effective control and not nominal authority, and its account fails.

(EC)(a) is a negligence-type standard: awareness of the particular wrong is not required, only a position from which the risk could be recognised. Accounts of culpability differ on this point, some requiring awareness of wrongdoing at some earlier stage [94] and others reasonable foreseeability [72], as surveyed by Rudy-Hiller [68]; I side with the latter for answerability, which is weaker than blame, and on the former view (a) would not suffice for blame. (b) is the familiar requirement of a fair opportunity to avoid the wrong [29], read as effective control in the sense of the control gaps discussed by Hindriks and Veluwenkamp [31]. It is not 'ought implies can' in the sense of a principle about obligation; I use that principle separately in Section 8.5.

Two cases mark the ends of a range. In the first, the executor is told what to do and nothing about the norm's joint structure or about the others. It is not in a position to recognise the risk, so (EC)(a) fails and it is not answerable through its conduct. In the second, it is told the norm and that others may act. Now (a) holds, and (b) decides: it could have abstained. Whether abstaining is a reasonable expectation

of an executor that holds an express permission depends on its role and the stakes; in ordinary practice reliance on a permission from an authorised issuer is at least a partial excuse, though for artificial executors that needs its own argument. Between the cases lie executors that are told the norm but not that others act, or that others exist but not the norm, or that have a channel through which to ask. Inquiry is owed where there is reason to suspect a joint structure, and the analysis then runs with the enlarged information state. The exemption of executors therefore rests on what those who instructed them supplied. Where the whole policy is in the system prompt, as it can be, the second case is the default, and the conclusion is not that executors are exempt but that whether they could reasonably have abstained is a question the theory does not settle. In both cases the issuance defect remains: in the first the issuer did not supply what the executor needed, and in the second it gave the executor a permission that, in the joint context, it could not safely use.

A dilemma stands behind the word 'executor'. If executors are mandate-following automata, blame is inapplicable to them, and (EC) exempts them idly; if they are agents that can be told the norm and can reason about it, the model does not say what they can infer. I do not settle whether artificial agents can be answerable. The results hold whatever the executors are: they show that no executor left its permission or misread it. Where executors cannot be answerable, the question of answerability for what they do is put by (PR) below to those who fixed their policy.

8.3 Issuers and the bridge from nodes to persons

In the model the issuers are nodes, which may be artificial. In the world they are persons and organisations, and the bridge between them needs premises. Consider the parties around a delegation of the exclusion norm in the running example (Table 4): the principal who owns the norm; the integrator or deployer who chose the architecture, including whether sub-agents run in parallel and whether a gate exists; the provider of the orchestration framework, whose defaults may spawn sub-agents in parallel without shared state; the provider of the model, whose dispositions determine how it issues and reads mandates; and a human who approves plans. I use two premises.

(PR) Policy responsibility. An issuance made by an artificial node is, for the purposes of answerability, an issuance by those who fixed the policy by which the node issues (its prompt, model and framework), to the extent that they were in a position to foresee the class of norms it would face. This is a premise about derivative responsibility of the kind that principal–agent accounts of automated systems invoke [58]. It addresses a structural gap, unsound delegation by agents that do what they are told, and not the gap that arises when operators cannot foresee the behaviour of a learning system [49].

(FR) Form recognition. Whether a norm is a product across a split, and hence whether static parallel delegation of it can be exact, is recognisable to those who fix a delegation, or the policy by which agents generate one, from the type of the norm (a shared budget, an exclusion, a duty, an ordering), without a search.

(FR) is a claim about what those parties are in a position to recognise, not about what they know. It is defeasible: for a norm generated at run time, or one whose structure is buried in natural language, the form may not be recognisable. Together, (PR) and (FR) do not select 'the nearest human'. Answerability for the design follows those

who had control over the schedule and interface and were in a position to recognise the form of the norm; a human who approves a plan generated at run time, and who sees neither the norm's structure nor the schedule, satisfies neither. That is what distinguishes the criterion from a moral crumple zone [25], provided (EC)(b) is read as effective authority to act: a deployer that shows a human the structure while giving the human neither time nor authority to act would only move the failure from ignorance to nominal knowledge. Nor is the principal exempt by construction: the specification of N is the principal's, and a mis-specification is a defect outside the tree that the model does not represent and the theory does not excuse.

Table 4 The exclusion norm of the running example, delegated to two payers in parallel with no gate. For each party, what it fixed, its position with respect to (FR) and (EC)(a), and what Proposition 11 yields. The assignments are illustrative and depend on the facts of a case.

Party	What it fixed	Position	Yield
Payer agents	Own behaviour within a permission	Told only to pay; cannot see the other	Not answerable through conduct (first case of the executor analysis)
Integrator or deployer	Architecture: parallel schedule, no gate; the instructions the agents receive	Exclusion is a shared budget; its form is recognisable	Answerable for the design
Principal	The norm, correctly stated here; possibly the architecture	Can recognise the form of the norm it states	Answerable for the decision to delegate this norm to this architecture only if it made that decision
Framework provider	Defaults for spawning and shared state	A general-purpose default does not foresee this class of norm	Not answerable under (PR), unless the class was foreseeable
Model provider	Dispositions in issuing and reading mandates	A general-purpose model does not foresee this class of norm	Not answerable under (PR) here
Human approving the plan	Approval of a plan generated at run time	Sees neither structure nor schedule; no effective authority	Not answerable for the design as things stand; if shown the structure it would be answerable, but not blameworthy without effective authority

8.4 The argument

Two further premises connect an executor's exemption and an issuer's position to an attribution. The first is a sufficiency principle, which (EC) does not supply because it states only necessary conditions; the second is the tracing of responsibility to earlier choices that produced later ignorance or inability [26], in a form that requires the earlier choice itself to satisfy an epistemic condition, a requirement that limits how far tracing reaches and that critics of tracing press hard [85].

(AN) Answerability. A party that by its own choices fixed a determinant of a violation (the mandates, the schedule or the interface) and satisfies (EC)(a) with respect to that choice is answerable for having exposed the norm to that risk and, when the

violation occurs, for the violation. This is the familiar principle that those who fix the terms of an undertaking and can see that it risks a bad outcome are owed a demand for an account of it; it does not say that they are to blame. It can give a demanding answer: a deployer who could recognise a small but non-negligible hazard is answerable even if accepting it was reasonable, and the account may then succeed; that is why answerability is not blame.

(TR) Tracing. If a party's choice put another out of the position in (EC)(a) or deprived it of the opportunity in (EC)(b), and the first party itself satisfied (EC)(a) with respect to that choice, then the answerability that the second party's exemption leaves open traces to the first.

Proposition 11 (Conditional answerability for the design, as an argument from premises) *Suppose (EC), (AN), (TR), (PR) and (FR). Consider a violation of N under a static delegation in which no node has an execution defect and no executor was in a position to recognise that its behaviour risked contributing. Then:*

(i) no executor is answerable through its conduct;

(ii) some node v has an issuance defect;

(iii) if M_v is not a product across the split at v, so that no static issuance at v could have been both sound and lossless, every party that fixed the policy for that issuance, could foresee the class of norms the node would face and could recognise the form of the norm is answerable, together with the others, for having exposed the norm to the risk and, the violation having occurred, for the violation;

(iv) if such a party was not in that position because of a choice by another party that itself satisfied (EC)(a) with respect to it, answerability traces to that party.

Derivation. (i) By hypothesis (EC)(a) fails for every executor, and (EC) states it as necessary. (ii) No node has an execution defect and the norm is violated, so some node has an issuance defect (Theorem 8). (iii) By (PR) the issuance at v is, for answerability, an issuance by those who fixed its policy, to the extent that they could foresee the class of norms the node would face. By (FR) those who could recognise the form of the norm could recognise, from Corollary 9(b), that a permissive static parallel issuance of it is unsound or lossy, that is, that it carries a non-negligible risk of unsoundness unless a loss is accepted (Table 3 gives the hazards of the norm families under uniform draws, from 1.6% upwards), so (EC)(a) holds for them with respect to the issuance, and by (AN) they are answerable. If M_v is a product across the split, a sound and lossless issuance existed, and answerability turns on the ordinary standard of diligence in issuing it. Several parties may qualify, and the argument does not apportion answerability among them. (iv) is (TR).

The proposition is an argument and not a theorem of the model. Its structural inputs are Theorem 8 and Corollary 9(b); Lemma 10 explains why an executor's exemption depends on what it was told and not on the norm alone; its normative inputs are (EC), (AN), (TR), (PR) and (FR). Its scope is narrower than it may read. (FR) will often fail for a node whose mandate is a sub-mandate generated at run time, and the proposition then bites mainly at the root's issuance and at architectures fixed at design time, and for run-time hierarchies at those who fixed the policy that generates them; where

nobody who fixed the delegation could recognise the form of the norm, (iii) and (iv) find no one, and the theory identifies that as the case where a responsibility void can remain [11]. The proposition agrees with Kong [42] that answerability can attach at the decision to delegate; it differs where (FR) fails, since Kong holds that answerability is conserved despite epistemic remoteness while the proposition leaves a possible void. It is compatible with the view that responsibility for a violation by compliant agents may lie in the procedure by which a group acts and not in any member's conduct [46, 61], which offers a rival locus (the insurer as a corporate agent) and not a corollary.

8.5 Securing, exactness and proportionality

Securing is not exactness. A safe issuance secures the norm; an exact one secures it at no loss of capability. For a non-product norm, static exact delegation is impossible (Corollary 9), but safe static delegation always exists: restrict, or designate a bearer. A principal can therefore always secure a norm by delegation, at a price, provided the agents comply and read their mandates faithfully, and only unsafe issuances fail to secure it. I do not claim that a duty is non-delegable or that a norm cannot be discharged by delegation. The claim is about a design fact: a norm with joint structure cannot be secured by static parallel delegation without loss.

An informational 'ought implies can'. The principle applies to obligations, and the results supply an instance. If the interface cannot carry the residual (Theorem 3, Corollary 4), no one who is obliged both to secure the norm and to complete the principal's task through that interface can do both. If 'ought implies can' is accepted, the obligation must be revised: change the interface, accept the loss, or do not delegate. This applies to issuers as much as to executors: an issuer whose schedule and interface are fixed by others, for instance by a framework that runs sub-agents in parallel with no shared state, faces a dilemma and is not defective for choosing the loss. Those who reject the principle can read the same result as a dilemma with a remainder [48, 88], an analogy that holds only to the extent that the capability lost is itself morally required, for instance by the principal's task.

Proportionality. An unsafe issuance is a risk imposition, and blame for it depends on the magnitude of the hazard, the stakes, the availability and cost of alternatives, and the consent of those affected. The cost profile supplies the alternatives and their prices, and the hazard of Table 3 ranges from 1.6% to 96.2% under uniform draws, a property of the norm under an assumed distribution and not an estimate of real risk. An unsafe profile for a low-stakes norm at a low hazard may be reasonable, and for a critical norm at a high hazard it may not. The theory does not say how the norm should be weighed against the capability the safe alternatives forgo; that belongs to the principal and those affected, and a principal that has decided to retain a safe loss has not defaulted on the norm. This is where (EC)(b) and the account of a party who could justify its choice do their work, and it is why (AN) yields answerability and not blame.

Risk and outcome. Issuance defects are defined before anything runs, independently of whether a violation occurs. Answerability for having exposed the norm to the risk does not depend on the outcome; answerability and blame for the violation as an outcome may, which is the familiar problem of moral luck. *Many hands.* Several nodes and several parties may be involved. The results identify the defects, and Corollary 9

identifies where an issuance defect or a loss is forced, but they do not apportion answerability among the parties. The observation of Kalluri [38] that responsibility may be shared across initiation, checkpoint, delegation and outcome is compatible with what is claimed here. *Concentration.* A bearer of last resort, as in Theorem 7, is a duty-discharger inside the hierarchy. It is not the person who accepts responsibility in advance in proposals to bridge responsibility gaps [17]. Concentrating responsibility in such a person has costs: Naumov and Tao [53] show that, under perfect information, decision mechanisms without responsibility gaps are elected dictatorships, and a designation that does not give the person the residual and the authority to act only creates a scapegoat.

8.6 Objections and rivals

The operative mandate is unobservable. The split between misreading and mis-refinement requires it, and audits cannot see it. The issuance defect, however, depends only on the assigned and issued mandates, and those can be recorded together with the state carried, where mandates are structured enough to be recorded and compared. A log of actions alone cannot separate an execution defect from an issuance defect, so proposals for activity logs [18] would need to record more. *Issuers cannot check inclusions in large spaces.* (FR) requires recognising the form of the norm, not checking an inclusion; where the norm is too complex for even that, the void described above is a possibility and I do not deny it. *Complicity.* On views on which those who take part intentionally in a joint endeavour are accountable for the harm it causes, whatever difference their own contribution made [45], executors that took part in it are answerable after all and clause (i) fails; and on a theory on which any agent that causally contributes is answerable irrespective of what it could know, the attribution differs again. The formal decomposition then shows what the disagreement is about. *Nothing here is specific to AI.* As to structure, agreed: the model applies to departments and subcontractors. Four things may make the matter more acute for agents: the number and speed of delegations, hierarchies generated at run time, mandates in natural language whose operative reading is opaque, and executors that may not be candidates for answerability at all, so that the question cannot be deflected onto them. *Law.* Design-level responsibility that is not discharged by delegating to competent staff has a familiar legal analogue in an employer's personal duty to provide a proper system of work [90]. It is an analogue for the location of design-level responsibility and not for its conditions: the employer's duty is a rule of liability that does not depend on the employer's awareness, so it is stricter than (EC), and liability is outside what I consider.

9 Implications and limits

9.1 Design and governance implications

Three consequences follow, each an application of a result and none a general principle.

Check the form of the norm before delegating. For each norm, decide whether it is a product; if not, decide which of the four moves of Section 3 the delegation will use, and record the choice. A product norm can be delegated to local mandates in any

hierarchy (Theorem 7(a)). A shared budget, an exclusion, a duty or an ordering cannot be delegated to parallel agents that see nothing of one another without a loss or a risk, and Table 3 says what each alternative costs. Where coupled agents can be placed on lines of authority, so that an ancestor resolves the coupling before the descendants run, sibling subtrees can work in parallel (Theorem 5(c)); the safe alternatives to a gate are designation, restriction and sequencing.

Keep aggregate, exclusive and collective norms at a state-holding point, or name a bearer. A single monitor or gate is the carriage of Theorem 3 at one node. It is the standard engineering answer, and the results describe what it must carry and what a cheaper design forgoes: without it the delegator accepts the loss of Theorem 7 or the hazard of Proposition 2. A collective duty needs a designated bearer or a bearer of last resort whose obligation is triggered by carried state; 'all agents may escalate' delivers nothing. Table 5 reads the common mechanisms at their strongest against the results. The theory adds little for a practitioner who already keeps a cap in a database: it says which norms need such a mechanism, what it must carry, what a cheaper design forgoes, and what follows for answerability when it is absent.

Record what was assigned, issued and carried. An issuance defect can be found before anything runs by checking an inclusion, and afterwards from records of the assigned and issued mandates and of the state carried, where mandates are structured enough to be recorded and compared; a log of actions alone cannot separate an execution defect from an issuance defect. Proposals for activity logs and agent identifiers [18] would need to be extended in this direction. Where a norm has joint structure, an overseer might be better served by being shown the residual than the transcript, where it can be computed, and a duty owed to 'the humans' needs a named bearer who is given it.

Table 5 Common mechanisms read at their strongest against the results of this paper.

Mechanism	What it secures	Cost and residual limit
Per-agent filters, system prompts, scoped credentials [6, 86]	Every product-form norm, exactly (Proposition 1)	Hazard $\eta(N) > 0$ for any non-product norm, however good the filter (Proposition 2)
Static quotas, escrow, designated bearer [4, 59]	Any norm, safely; with known demand, without loss of the wanted plan	Loses capability when demand is unknown at issuance (Theorem 7); cannot follow a run
Transactional gate: unique index, atomic counter, policy engine with session state [3, 13]	Any norm the gate can evaluate, exactly; it is carriage at one node (Theorem 3)	Serialised commits, a component to trust, and coverage of every act; a bypass defeats it
Workflow or orchestrator sequencing	Any norm, exactly, on an N-independent schedule	$d(N)$ sequential decision rounds (Theorem 5)
Human oversight	Whatever the overseer can evaluate	May be better served by the residual than the transcript, where computable; a duty needs a named bearer

9.2 Limitations

The model is extensional and cannot state norms about reasons (Example 2); it is finite, deterministic and has a fixed agent set, and treats stochastic environments, run-time spawning and several principals only as extensions (Table 1). Issuance in the defect

theory is static; the carriage and schedule theory allows reactive mandates but lets them depend on the whole earlier history, as a coordinator or gate would, so I give no per-edge bound and no result that ties the two halves together beyond Corollary 9. Whether a norm is a product across a split depends on how agents are individuated: merge two payers into one and the exclusion becomes a product. That individuation is itself a design decision, and one for which a party is answerable. The counting measures of the cost profile are choices, and the retention measure describes a delegator that does not know which plan will be wanted. The depth and carriage computations are exponential in the number of agents. The answerability argument rests on premises that some will reject: a theory on which any agent that causally contributes is answerable irrespective of what it could know will attribute differently, and the formal decomposition then shows what the disagreement is about. The illustrative comparison of architectures rests on an assumed error model, and no claim is made about how deployed language-model agents misread state or about how often the failures analysed here occur.

9.3 Future research

Five directions follow. (1) A per-edge theory of carriage for trees in which information travels only along delegation edges, and a bridge between static defects and reactive schemes. (2) A stochastic theory, in which the residual becomes a set of conditional continuations and exactness a bound on the probability of violation, and an approximate theory that trades bits against loss. (3) Structured norm representations, such as constraint languages or policy languages, for which the profile can be computed at scale. (4) Empirical study of how often language-model agents preserve stateful constraints across hand-offs, using tasks whose norms have known cost profiles, and analysis of documented incidents in the terms used here. (5) Multi-principal and pluralistic norms, and the responsibility questions that follow when the intersection of admissible sets is empty.

10 Conclusion

Local compliance secures a norm only when the norm is a product of individual requirements, a familiar fact that I have restated for delegation. For every other norm, some behaviour of a hierarchy of fully compliant agents violates it, and what must be added is fixed by the norm's residual structure: the state that must cross each cut of the hierarchy and the number of sequential stages that parallel delegation needs. Both can be computed for explicitly given norms, and they follow the shape of the coupling and not its amount: n ordered slots need $\lceil \log_2(n+1) \rceil$ stages of concurrent decisions, one shared unit needs n, and a collective duty needs two stages and one bit, in a model without a shared atomic object. When no executor was in a position to recognise the risk of its contribution, no executor is answerable through its conduct and no execution defect explains the violation, and on the premises stated answerability for the design of the delegation lies with those who fixed the mandates, the schedule and the interface and could have recognised the form of the norm. That is not a claim of blame: safe but lossy delegation is always available, whether to accept a loss is a matter of proportion, and where no one who fixed the delegation could recognise the form of the norm, a void

may remain. The results are structural. They say nothing about how often deployed agents misread their hand-offs, and they suggest, and do not prove, what to check before a norm is delegated, what a hand-off must carry, and where to look when a hierarchy of agents that behaved as instructed does not deliver what the principal required.

Acknowledgements. The author thanks the Indian Knowledge Systems and Mental Health Applications (IKSMHA) Centre, Indian Institute of Technology Mandi, for institutional support.

Declarations

Funding. No funding was received for conducting this study.

Competing interests. The author has no competing interests to declare that are relevant to the content of this article.

Author contributions. Lolugu Vasudeva Prabhath is the sole author and was responsible for conceptualisation, methodology, formal analysis, software, investigation, visualisation, and writing (original draft, and review and editing).

Ethics approval and consent. Not applicable. The study involved no human participants, human data or animals.

Data availability. All results reported in this paper were generated by the code provided as Online Resource 2 and are provided with it as result files (Online Resources 1 and 2). No external datasets were used.

Code availability. The code needed to regenerate all computations, tables and figures is provided as Online Resource 2, with fixed seeds and an environment specification.

Use of generative AI. Generative AI (Claude, Anthropic) was used in preparing this work: for literature search and summarisation, for drafting and revising text and LaTeX, for proposing and checking proofs, and for writing and debugging the analysis code. The author is responsible for the content of the paper, including the proofs, the code, the results and the references.

Appendix A Proofs

Throughout, $\pi_i(N)$ is the projection of N on agent i and $\mathrm{hull}(N) = \prod_i \pi_i(N) \supseteq N$.

Proof of Proposition 1.

(a)⇔(b) is the definition of exactness, with $M_i = N_i$; an exact profile forces $N = \prod_i M_i$. (a)⇒(c) is immediate. For (c)⇒(a), let $y \in \mathrm{hull}(N)$ and choose $x^i \in N$ with $x^i_i = y_i$; starting from $z = x^1$ and applying (c) with $x^2, \ldots, x^n$ in turn replaces one coordinate at a time and leaves $z = y \in N$, so $\mathrm{hull}(N) \subseteq N$. This also gives (a)⇔(e), since $\mathrm{hull}(N) \supseteq N$ with equality iff N is a product. For (c)⇔(d): if $y \in N/x_i$ and $x' \in N$ then $(x_i, y) \in N$ and (c) gives $(x'_i, y) \in N$, so all live behaviours of i have the same residual; conversely, equal residuals give (c).

Proof of Proposition 2.

(a) A complete profile has $M_i \supseteq \pi_i(N)$ for all i, so $[\![m]\!] \supseteq \mathrm{hull}(N)$; by Proposition 1(e), $\mathrm{hull}(N) \setminus N \neq \varnothing$ when N is not a product. (b) A safe profile that lost nothing would satisfy $[\![m]\!] = N$, that is, be exact, contradicting Proposition 1(b). (c) Independent uniform draws from the M_i are uniform on $[\![m]\!]$, and $|[\![m]\!] \setminus N|/|[\![m]\!]| = 1 - |N|/|[\![m]\!]| \geq 1 - |N|/|\mathrm{hull}(N)| = \eta(N)$, because $N \subseteq \mathrm{hull}(N) \subseteq [\![m]\!]$. The projection profile has $[\![m]\!] = \mathrm{hull}(N)$ and attains the bound.

Proof of Theorem 3.

(a) Let Σ be exact and $k < d$. By completeness, every live $h \in X_{\leq k}$ is consistent with Σ: any $x \in N$ extending h is a run. Let h, h' be live with $N/h \neq N/h'$; without loss of generality there is $y \in N/h \setminus N/h'$. Then $(h, y) \in N$ is a run, and $(h', y) \notin N$ is not a run, by safety. Let $j > k$ be the least wave at which (h', y) leaves the mandates: there is $i \in W_j$ with $y_i \notin M_i^j(h', g)$, where g is the part of y in waves $k{+}1, \ldots, j{-}1$. Since (h, y) is a run, $y_i \in M_i^j(h, g)$. Hence $\mu_h(j, g) \neq \mu_{h'}(j, g)$ and $\mu_h \neq \mu_{h'}$. Distinct residuals give distinct mandate maps, so $c_k(\Sigma) \geq \rho_{\leq k}(N)$.

(b) Let σ be N-independent and define $M_i^{k+1}(h) = \pi_i(N/h)$ for $i \in W_{k+1}$ and live $h \in X_{\leq k}$; it is non-empty because h is live. For non-live h let $M_i^{k+1}(h) = X_i$. The live set of wave $k{+}1$ after a live h is $\pi_{W_{k+1}}(N/h)$, a product by independence, hence equal to $\prod_i M_i^{k+1}(h)$. By induction every prefix of a run is live: this holds for $k = 0$ because $N \neq \varnothing$, and if $x_{\leq k}$ is live and $x_{W_{k+1}} \in \prod_i M_i^{k+1}(x_{\leq k}) = \mathrm{Live}_{k+1}(x_{\leq k})$ then $x_{\leq k+1}$ is live. At $k = d$ liveness is membership in N, so every run lies in N. Conversely if $x \in N$ every prefix is live and $x_{W_{k+1}} \in \mathrm{Live}_{k+1}(x_{\leq k})$, so x is a run. The mandate map after h depends on h only through N/h, because $N/(h, g) = N/h/g$; so $c_k \leq \rho_{\leq k}(N)$, and (a) gives equality.

Proof of Corollary 4.

If the later mandates depend on h only through $q(h)$, then $q(h) = q(h')$ implies $\mu_h = \mu_{h'}$, and the proof of Theorem 3(a) shows that an exact scheme has $\mu_h \neq \mu_{h'}$ for live histories with different residuals. The same holds if the mandates depend on $q(h)$ only through $f(q(h))$, since $f(q(h)) = f(q(h'))$ again implies $\mu_h = \mu_{h'}$.

Proof of Theorem 5.

(a) If σ is N-independent, Theorem 3(b) gives an exact scheme. Conversely let Σ be exact, $k \leq d$, and h a live history in $X_{<k}$; it is consistent by completeness. Let $R = \prod_{i \in W_k} M_i^k(h)$. If $a \in \mathrm{Live}_k(h)$, some $x \in N$ extends (h, a) and is a run, so $a \in R$. If $a \in R$, since all later mandates are non-empty there is a run extending (h, a), which lies in N by safety, so $a \in \mathrm{Live}_k(h)$. Hence $\mathrm{Live}_k(h) = R$ is a product.

(b) A wave of one agent is always independent, so the schedule of singletons gives $d \leq n$. The only one-wave schedule is (V), and $\mathrm{Live}_1(\varnothing) = N$, so $d = 1$ iff N is a product. Let $N = N' \times N''$ on $V' \sqcup V''$. For a live history (h', h'') the live set of a wave W is $\mathrm{Live}'(h') \times \mathrm{Live}''(h'')$ with factors concerning $W \cap V'$ and $W \cap V''$; a product of non-empty sets is a product over all coordinates iff each factor is. An N-independent schedule therefore restricts to independent schedules for N' and N'', giving

$d(N) \geq \max\{d(N'), d(N'')\}$; independent schedules for the factors, padded to equal length and merged wave by wave, are N-independent, giving the reverse inequality.

(c) Let F be a rooted forest on V in which every edge of G joins a node to one of its ancestors; a forest of least height has height $\mathrm{td}(G)$. The scope of each constraint is a clique of G, and any two nodes of a clique are comparable in F, so the scope lies on a single chain of F. Let W_k be the nodes of depth k and let h be a live history over the nodes of depth less than k. For $u \in W_k$ let T_u be its subtree. A chain that meets depth $\geq k$ meets it inside T_u for exactly one $u \in W_k$, and its nodes of smaller depth are fixed by h. Hence, given h, every constraint is either fixed or concerns the behaviours of a single T_u. The condition that (h, a) can be completed to a member of N therefore splits into independent conditions on a_u, one for each $u \in W_k$, each involving only T_u and h. So $\mathrm{Live}_k(h) = \prod_{u \in W_k} P_u(h)$, the level schedule is N-independent, and it has $\mathrm{td}(G)$ waves; thus $d(N) \leq \mathrm{td}(G)$.

(d) Cap. Let $N = \{\sum_i x_i \leq B\}$, $1 \leq B < n$. Consider a wave W of $w \geq 2$ agents after a set A of a agents. For a live history with remaining budget r the live set is $\{u \in \{0,1\}^W : |u| \leq r\}$, since later agents can abstain. This is a product iff $r \geq w$ or $r = 0$. The reachable r are $B - t$ for $0 \leq t \leq \min(a, B)$. If $a \geq B - 1$ then $r = 1$ is reachable and $1 < w$, so the wave is not independent; if $a < B - 1$ the least reachable r is $B - a \geq 2$ and independence holds iff $a + w \leq B$. Thus a wave of two or more agents is independent iff at most B agents have acted by its end. Agents in such waves number at most B and all others are in singleton waves, so any independent schedule has at least $1 + (n - B)$ waves; the schedule with B agents in the first wave and the rest singly attains it.

(d) Duty. For $n \geq 2$ the norm is not a product, so $d \geq 2$. The schedule with all agents but one, β, in the first wave and β in the second is independent: after the first wave β can still discharge the duty, so every first-wave behaviour is live and $\mathrm{Live}_1(\varnothing)$ is the full space; β alone is a singleton.

(d) Walls. The norm is a product of two-agent non-product norms, each of depth 2, so $d = 2$ by (b).

(d) Ordering. Upper bound: the path on n vertices has treedepth $\lceil \log_2(n+1) \rceil$ and the ordering norm is a conjunction of binary constraints on it, so (c) applies. Lower bound: let D_n be the depth of the chain on n agents, $D_0 = 0$, $D_1 = 1$. For $n \geq 2$ and $K \geq 2$, two agents $i < j$ in the first wave would have live set $\{(a_i, a_j) : a_i \leq a_j\}$, not a product; so the first wave is a single agent v. The history $x_v = K - 1$ is live and leaves a free chain of $v - 1$ agents to the left and a pinned right side; the history $x_v = 0$ leaves a free chain of $n - v$ agents on the right and a pinned left side. A schedule independent for both restricts to independent schedules for both free chains (by the product argument of (b)), so $d \geq 1 + \max\{D_{v-1}, D_{n-v}\}$. Minimising over v gives $D_n \geq 1 + \min_v \max\{D_{v-1}, D_{n-v}\} = \lceil \log_2(n+1) \rceil$ by induction.

Proof of Proposition 6.

Let Σ be a safe scheme for a schedule with w waves; its runs lie in $N = \{0\} \cup \{e_1, \ldots, e_n\}$. Suppose two distinct unit vectors e_i, e_j are runs with i, j in the same wave W_k. Their common prefix before wave k is the zero history, so $1 \in M_i^k(0)$ and $1 \in M_j^k(0)$, and $M^k(0)$ is a rectangle: the behaviour that sets both to 1 and every other agent of the

wave to 0 is compliant and, since later mandates are non-empty, extends to a run of weight 2, which lies outside N, contradicting safety. So each wave contributes at most one unit-vector run and there are at most $w+1$ runs. Conversely, pick one agent i_k per wave, let i_k range over $\{0,1\}$ and every other agent of the wave over $\{0\}$ as long as no 1 has occurred, and let everyone range over $\{0\}$ afterwards; the runs are 0 and $e_{i_1},\dots,e_{i_w}$. Hence the greatest retention is $(w+1)/(n+1)$.

Proof of Theorem 7.

(a) follows from Proposition 1 and Theorem 5(b). (b)(i) Let m be safe and suppose every M_i, $i \in Q$, contains some $y_i \notin D_i$. The joint behaviour with y_i at each $i \in Q$ (and any element of M_j elsewhere) is compliant and outside N_D, contradicting safety. So some $M_i \subseteq D_i$, and m is contained in the designation of i, which is safe because $x_i \in D_i$ on its compliance set. Since $|Q| \geq 2$ and each $D_j \neq \varnothing$, there is an admissible joint behaviour with $x_i \notin D_i$ (take $x_j \in D_j$ for some $j \neq i$), which is not compliant with $M_i \subseteq D_i$; so no safe profile is complete. The designations are pairwise incomparable, so they are the maximal safe profiles. (b)(ii) After a history in which some agent of Q has behaved within its D_i the residual is unconstrained; after one in which none has, the residual requires β to behave within D_β. Every first-wave behaviour is live because β can still discharge the duty, so the first wave is independent and the residual scheme is exact by Theorem 3(b); two residuals occur. (c) Safety of a static profile on $\{0,1\}^n$ means that the largest compliant total, the number of agents with $1 \in M_i$, is at most B. The size of the compliance set is maximised by taking exactly B agents with $M_i = \{0,1\}$ and the rest with $M_i = \{0\}$, giving 2^B; and $|N_B| = \sum_{j \leq B} \binom{n}{j}$. For the continuous case let b_i be the essential supremum of M_i. If $\sum_i b_i > 1$, choose $\delta > 0$ with $\sum_i (b_i - \delta) > 1$; the sets $M_i \cap (b_i - \delta, b_i]$ have positive measure and their product, of positive measure, lies in $\{\sum x_i > 1\}$, contradicting safety. So $\sum_i b_i \leq 1$ and $\mathrm{vol}(\prod_i M_i) \leq \prod_i b_i \leq n^{-n}$ by the arithmetic–geometric mean inequality, while $\mathrm{vol}(N) = 1/n!$. The ratio is $n!/n^n$.

Proof of Theorem 8.

Suppose no node has an execution or an issuance defect. I show by induction from the leaves that $x_{S_v} \in M_v$ for every v. For a node v, $x_v \in L_v$ and, for each child c, $x_{S_c} \in M_c$ by the induction hypothesis; hence $x_{S_v} \in L_v \times \prod_c M_c \subseteq M_v$, the inclusion being the absence of an issuance defect at v. At the root, $x \in M_r = N$. So $x \notin N$ implies a defect of one of the two kinds. An issuance defect at v implies an interpretation or a refinement defect at v: if $\dot{M}_v \subseteq M_v$ and $L_v \times \prod_c M_c \subseteq \dot{M}_v$, then $L_v \times \prod_c M_c \subseteq M_v$.

Proof of Corollary 9.

(a) Unfolding $M_r = N = L_r \times \prod_c M_c$ and $M_c = L_c \times \prod M_{c'}$ from the root down gives $N = \prod_v L_v$. (b) An issuance at v is a rectangle $L_v \times \prod_c M_c$ in the space $X_v \times \prod_c X_{S_c}$; it is exact iff it equals M_v, which by Proposition 1 is impossible when M_v is not a product across that split. So the rectangle is not contained in M_v, an issuance defect, or is properly contained in it, a loss.

Proof of Lemma 10.

Fix $y \in [\![m]\!] \setminus N$ and an agent i, and let $I_i = \prod_{j \neq i} M_j$. Then $b' = y_{-i} \in I_i$ and $(y_i, b') = y \notin N$. Since $y_i \in M_i$ is live under m, there is $b \in I_i$ with $(y_i, b) \in N$. Hence $(y_i, b) \in N$ for some $b \in I_i$ and $(y_i, b') \notin N$ for another, so the contribution of y_i is not determined by I_i. The projection profile satisfies the hypothesis: for $a \in \pi_i(N)$ choose $x \in N$ with $x_i = a$; then $x_{-i} \in \prod_{j \neq i} \pi_j(N)$.

References

[1] Adalat O, Hamel-De le Court E, Belardinelli F (2026) Contract-based compositional shielding for safe multi-agent reinforcement learning. arXiv preprint arXiv:2606.14130, URL https://arxiv.org/abs/2606.14130

[2] Ågotnes T, van der Hoek W, Wooldridge M (2010) Robust normative systems and a logic of norm compliance. Logic Journal of the IGPL 18(1):4–30. https://doi.org/10.1093/jigpal/jzp070

[3] Bailis P, Fekete A, Franklin MJ, Ghodsi A, Hellerstein JM, Stoica I (2014) Coordination avoidance in database systems. Proceedings of the VLDB Endowment 8(3):185–196. https://doi.org/10.14778/2735508.2735509

[4] Barbará-Millá D, Garcia-Molina H (1994) The demarcation protocol: a technique for maintaining constraints in distributed database systems. The VLDB Journal 3(3):325–353. https://doi.org/10.1007/BF01232643

[5] Bauer A, Falcone Y (2016) Decentralised LTL monitoring. Formal Methods in System Design 48(1):46–93. https://doi.org/10.1007/s10703-016-0253-8

[6] Birgisson A, Politz JG, Erlingsson U, Taly A, Vrable M, Lentczner M (2014) Macaroons: cookies with contextual caveats for decentralized authorization in the cloud. In: Network and Distributed System Security Symposium (NDSS 2014), URL https://www.ndss-symposium.org/ndss2014/ndss-2014-programme/macaroons-cookies-contextual-caveats-decentralized-authorization-cloud/

[7] Bodlaender HL, Gilbert JR, Hafsteinsson H, Kloks T (1995) Approximating treewidth, pathwidth, frontsize, and shortest elimination tree. Journal of Algorithms 18(2):238–255. https://doi.org/10.1006/jagm.1995.1009

[8] Boella G, van der Torre L, Verhagen H (2006) Introduction to normative multiagent systems. Computational & Mathematical Organization Theory 12(2–3):71–79. https://doi.org/10.1007/s10588-006-9537-7

[9] Bovens M (1998) The Quest for Responsibility: Accountability and Citizenship in Complex Organisations. Theories of Institutional Design, Cambridge University Press

[10] Bovens M (2007) Analysing and assessing accountability: A conceptual framework. European Law Journal 13(4):447–468. https://doi.org/10.1111/j.1468-0386.2007.00378.x

[11] Braham M, van Hees M (2011) Responsibility voids. The Philosophical Quarterly 61(242):6–15. https://doi.org/10.1111/j.1467-9213.2010.677.x

[12] Braham M, van Hees M (2012) An anatomy of moral responsibility. Mind 121(483):601–634. https://doi.org/10.1093/mind/fzs081

[13] Brewer DFC, Nash MJ (1989) The Chinese Wall security policy. In: Proceedings of the 1989 IEEE Symposium on Security and Privacy, pp 206–214, URL https://doi.org/10.1109/SECPRI.1989.36295

[14] Brobbey AA, Blu R (2026) Consent collapse in agentic AI: moral authorization beyond the boundaries of task delegation. AI and Ethics 6(4):431. https://doi.org/10.1007/s43681-026-01287-w

[15] Bryant RE (1986) Graph-based algorithms for Boolean function manipulation. IEEE Transactions on Computers C-35(8):677–691. https://doi.org/10.1109/TC.1986.1676819

[16] Cemri M, Pan MZ, Yang S, Agrawal LA, Chopra B, Tiwari R, Keutzer K, Parameswaran A, Klein D, Ramchandran K, Zaharia MA, Gonzalez JE, Stoica I (2025) Why do multi-agent LLM systems fail? In: Advances in Neural Information Processing Systems 38 (NeurIPS 2025), Datasets and Benchmarks Track, pp 135676–135729, URL https://doi.org/10.52202/085713-4082

[17] Champagne M, Tonkens R (2015) Bridging the responsibility gap in automated warfare. Philosophy & Technology 28(1):125–137. https://doi.org/10.1007/s13347-013-0138-3

[18] Chan A, Ezell C, Kaufmann M, Wei K, Hammond L, Bradley H, Bluemke E, Rajkumar N, Krueger D, Kolt N, Heim L, Anderljung M (2024) Visibility into AI agents. In: Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT '24), pp 958–973, URL https://doi.org/10.1145/3630106.3658948

[19] Chockler H, Halpern JY (2004) Responsibility and blame: a structural-model approach. Journal of Artificial Intelligence Research 22:93–115. https://doi.org/10.1613/jair.1391

[20] Clark DD, Wilson DR (1987) A comparison of commercial and military computer security policies. In: Proceedings of the 1987 IEEE Symposium on Security and Privacy, pp 184–194, URL https://doi.org/10.1109/SP.1987.10001

[21] Danaher J (2016) Robots, law and the retribution gap. Ethics and Information Technology 18(4):299–309. https://doi.org/10.1007/s10676-016-9403-3

[22] Debenedetti E, Shumailov I, Fan T, Hayes J, Carlini N, Fabian D, Kern C, Shi C, Terzis A, Tramèr F (2026) Defeating prompt injections by design. In: 2026 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML), pp 587–618, URL https://doi.org/10.1109/SaTML68715.2026.00040

[23] Duijf H, Tamminga A, Van De Putte F (2021) An impossibility result on methodological individualism. Philosophical Studies 178(12):4165–4185. https://doi.org/10.1007/s11098-021-01642-z

[24] Elden Å (2026) When responsibility cannot arise: algorithmic mediation and the preconditions of moral imputability. AI and Ethics 6(5):500. https://doi.org/10.1007/s43681-026-01359-x

[25] Elish MC (2019) Moral crumple zones: cautionary tales in human-robot interaction. Engaging Science, Technology, and Society 5:40–60. https://doi.org/10.17351/ests2019.260

[26] Fischer JM, Tognazzini NA (2009) The truth about tracing. Noûs 43(3):531–556. https://doi.org/10.1111/j.1468-0068.2009.00717.x

[27] Fitoussi D, Tennenholtz M (2000) Choosing social laws for multi-agent systems: minimality and simplicity. Artificial Intelligence 119(1–2):61–101. https://doi.org/10.1016/S0004-3702(00)00006-0

[28] Greenblatt R, Shlegeris B, Sachan K, Roger F (2024) AI control: improving safety despite intentional subversion. In: Proceedings of the 41st International Conference on Machine Learning, pp 16295–16336, URL https://proceedings.mlr.press/v235/greenblatt24a.html

[29] Hart HLA (1968) Punishment and Responsibility: Essays in the Philosophy of Law. Clarendon Press

[30] Hellerstein JM, Alvaro P (2020) Keeping CALM: when distributed consistency is easy. Communications of the ACM 63(9):72–81. https://doi.org/10.1145/3369736

[31] Hindriks F, Veluwenkamp H (2023) The risks of autonomous machines: from responsibility gaps to control gaps. Synthese 201(1):21. https://doi.org/10.1007/s11229-022-04001-5

[32] van der Hoek W, Roberts M, Wooldridge M (2007) Social laws in alternating time: effectiveness, feasibility, and synthesis. Synthese 156(1):1–19. https://doi.org/10.1007/s11229-006-9072-6

[33] Honda K, Yoshida N, Carbone M (2016) Multiparty asynchronous session types. Journal of the ACM 63(1):9:1–9:67. https://doi.org/10.1145/2827695

[34] Horty JF (2001) Agency and Deontic Logic. Oxford University Press, URL https://doi.org/10.1093/0195134613.001.0001

[35] Hu Y, Wang R (2026) When local monitors miss compositional harm: diagnosing distributed backdoors in multi-agent systems. arXiv preprint arXiv:2607.11751, URL https://arxiv.org/abs/2607.11751

[36] Ibrahim A, Li Y (2026) Overlaying governance: a compositional authorization framework for delegation and scope in agentic AI. arXiv preprint arXiv:2606.03518, URL https://arxiv.org/abs/2606.03518

[37] Jones E, Dragan A, Steinhardt J (2025) Adversaries can misuse combinations of safe models. In: Proceedings of the 42nd International Conference on Machine Learning, pp 28327–28349, URL https://proceedings.mlr.press/v267/jones25a.html

[38] Kalluri R (2026) The responsibility cascade: moral attribution and governance challenges in agentic AI systems. Digital Society 5(3):47. https://doi.org/10.1007/s44206-026-00288-w

[39] Kolt N (2026) Governing AI agents. Notre Dame Law Review 101(1):335. URL https://ndlawreview.org/governing-ai-agents/

[40] Komenda J, Masopust T (2017) Computation of controllable and coobservable sublanguages in decentralized supervisory control via communication. Discrete Event Dynamic Systems 27(4):585–608. https://doi.org/10.1007/s10626-017-0249-6

[41] Komenda J, Masopust T, van Schuppen JH (2012) Supervisory control synthesis of discrete-event systems using a coordination scheme. Automatica 48(2):247–254. https://doi.org/10.1016/j.automatica.2011.07.008

[42] Kong CY (2026) The delegation illusion: why deploying autonomous AI agents does not diminish principal responsibility. AI and Ethics 6(5):518. https://doi.org/10.1007/s43681-026-01383-x

[43] Königs P (2022) Artificial intelligence and responsibility gaps: what is the problem? Ethics and Information Technology 24(3):36. https://doi.org/10.1007/s10676-022-09643-0

[44] Kushilevitz E, Nisan N (1997) Communication Complexity. Cambridge University Press, URL https://doi.org/10.1017/CBO9780511574948

[45] Kutz C (2000) Complicity: Ethics and Law for a Collective Age. Cambridge University Press, URL https://doi.org/10.1017/CBO9780511663758

[46] List C, Pettit P (2011) Group Agency: The Possibility, Design, and Status of Corporate Agents. Oxford University Press, URL https://doi.org/10.1093/acprof:oso/9780199591565.001.0001

[47] Liu JWH (1990) The role of elimination trees in sparse factorization. SIAM Journal on Matrix Analysis and Applications 11(1):134–172. https://doi.org/10.1137/0611010

[48] Marcus RB (1980) Moral dilemmas and consistency. The Journal of Philosophy 77(3):121–136. https://doi.org/10.2307/2025665

[49] Matthias A (2004) The responsibility gap: ascribing responsibility for the actions of learning automata. Ethics and Information Technology 6(3):175–183. https://doi.org/10.1007/s10676-004-3422-1

[50] Melcer D, Amato C, Tripakis S (2022) Shield decentralization for safe multi-agent reinforcement learning. In: Advances in Neural Information Processing Systems 35 (NeurIPS 2022), URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/57444e14ecd9e2c8f603b4f012ce3811-Abstract-Conference.html

[51] Modi PJ, Shen WM, Tambe M, Yokoo M (2005) Adopt: asynchronous distributed constraint optimization with quality guarantees. Artificial Intelligence 161(1–2):149–180. https://doi.org/10.1016/j.artint.2004.09.003

[52] Muruaga X (2026) Bounded agents: delegation security for multi-agent AI systems. arXiv preprint arXiv:2608.15888, URL https://arxiv.org/abs/2608.15888, 2608.15888

[53] Naumov P, Tao J (2025) Responsibility gap in collective decision making. In: Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25), pp 4606–4614, URL https://doi.org/10.24963/ijcai.2025/513

[54] Nerode A (1958) Linear automaton transformations. Proceedings of the American Mathematical Society 9(4):541–544. https://doi.org/10.1090/S0002-9939-1958-0135681-9

[55] Nešetřil J, Ossona de Mendez P (2006) Tree-depth, subgraph coloring and homomorphism bounds. European Journal of Combinatorics 27(6):1022–1041. https://doi.org/10.1016/j.ejc.2005.01.010

[56] Nissenbaum H (1996) Accountability in a computerized society. Science and Engineering Ethics 2(1):25–42. https://doi.org/10.1007/BF02639315

[57] Norman TJ, Reed C (2010) A logic of delegation. Artificial Intelligence 174(1):51–71. https://doi.org/10.1016/j.artint.2009.10.001

[58] Nyholm S (2018) Attributing agency to automated systems: reflections on human–robot collaborations and responsibility-loci. Science and Engineering Ethics 24(4):1201–1219. https://doi.org/10.1007/s11948-017-9943-x

[59] O'Neil PE (1986) The escrow transactional method. ACM Transactions on Database Systems 11(4):405–430. https://doi.org/10.1145/7239.7265

[60] Patil K (2026) SentinelAgent: intent-verified delegation chains for securing federal multi-agent AI systems. arXiv preprint arXiv:2604.02767, URL https://arxiv.org/abs/2604.02767

[61] Pettit P (2007) Responsibility incorporated. Ethics 117(2):171–201. https://doi.org/10.1086/510695

[62] Pinisetty S, Pradhan A, Roop P, Tripakis S (2021) Compositional runtime enforcement revisited. Formal Methods in System Design 59(1–3):205–252. https://doi.org/10.1007/s10703-022-00401-y

[63] van de Poel I, Nihlén Fahlquist J, Doorn N, Zwart S, Royakkers L (2012) The problem of many hands: climate change as an example. Science and Engineering Ethics 18(1):49–67. https://doi.org/10.1007/s11948-011-9276-0

[64] Rabin MO, Scott D (1959) Finite automata and their decision problems. IBM Journal of Research and Development 3(2):114–125. https://doi.org/10.1147/rd.32.0114

[65] Ramadge PJG, Wonham WM (1989) The control of discrete event systems. Proceedings of the IEEE 77(1):81–98. https://doi.org/10.1109/5.21072

[66] Rubel A, Castro C, Pham A (2019) Agency laundering and information technologies. Ethical Theory and Moral Practice 22(4):1017–1041. https://doi.org/10.1007/s10677-019-10030-w

[67] Rudie K, Wonham WM (1992) Think globally, act locally: decentralized supervisory control. IEEE Transactions on Automatic Control 37(11):1692–1708. https://doi.org/10.1109/9.173140

[68] Rudy-Hiller F (2022) The Epistemic Condition for Moral Responsibility. The Stanford Encyclopedia of Philosophy, URL https://plato.stanford.edu/archives/win2022/entries/moral-responsibility-epistemic/, winter 2022 Edition, edited by Edward N. Zalta and Uri Nodelman. First published 12 September 2018; substantive revision 4 October 2022

[69] Sandhu RS, Coyne EJ, Feinstein HL, Youman CE (1996) Role-based access control models. Computer 29(2):38–47. https://doi.org/10.1109/2.485845

[70] Schneider FB (2000) Enforceable security policies. ACM Transactions on Information and System Security 3(1):30–50. https://doi.org/10.1145/353323.353382

[71] Sears D, Rudie K (2016) Minimal sensor activation and minimal communication in discrete-event systems. Discrete Event Dynamic Systems 26(2):295–349. https://doi.org/10.1007/s10626-015-0212-3

[72] Sher G (2009) Who Knew? Responsibility Without Awareness. Oxford University Press, URL https://doi.org/10.1093/acprof:oso/9780195389197.001.0001

[73] Shi T, He J, Wang Z, Li H, Wu L, Guo W, Song D (2025) Progent: securing AI agents with privilege control. arXiv preprint arXiv:2504.11703, URL https://arxiv.org/abs/2504.11703

[74] Shoemaker D (2011) Attributability, answerability, and accountability: Toward a wider theory of moral responsibility. Ethics 121(3):602–632. https://doi.org/10.1086/659003

[75] Shoham Y, Tennenholtz M (1995) On social laws for artificial agent societies: off-line design. Artificial Intelligence 73(1–2):231–252. https://doi.org/10.1016/0004-3702(94)00007-N

[76] Shukla S, Joshi H (2026) Fundamental limits of runtime policy enforcement in multi-agent AGI systems. In: Iklé M, Franz A, Kemendo A, Lowy H (eds) Artificial General Intelligence (AGI 2026), Lecture Notes in Computer Science, vol 16855. Springer, pp 292–302, URL https://doi.org/10.1007/978-3-032-33195-3_21

[77] Santoni de Sio F, van den Hoven J (2018) Meaningful human control over autonomous systems: a philosophical account. Frontiers in Robotics and AI 5:15. https://doi.org/10.3389/frobt.2018.00015

[78] Santoni de Sio F, Mecacci G (2021) Four responsibility gaps with artificial intelligence: why they matter and how to address them. Philosophy & Technology 34(4):1057–1084. https://doi.org/10.1007/s13347-021-00450-x

[79] Sparrow R (2007) Killer robots. Journal of Applied Philosophy 24(1):62–77. https://doi.org/10.1111/j.1468-5930.2007.00346.x

[80] Tamminga A, Hindriks F (2020) The irreducibility of collective obligations. Philosophical Studies 177(4):1085–1109. https://doi.org/10.1007/s11098-018-01236-2

[81] Taylor I (2024) Collective responsibility and artificial intelligence. Philosophy & Technology 37(1):27. https://doi.org/10.1007/s13347-024-00718-y

[82] Thompson DF (1980) Moral responsibility of public officials: the problem of many hands. American Political Science Review 74(4):905–916. https://doi.org/10.2307/1954312

[83] Tigard DW (2021) There is no techno-responsibility gap. Philosophy & Technology 34(3):589–607. https://doi.org/10.1007/s13347-020-00414-7

[84] Tomašev N, Franklin M, Osindero S (2026) Intelligent AI delegation. arXiv preprint arXiv:2602.11865, URL https://arxiv.org/abs/2602.11865, 2602.11865

[85] Vargas M (2005) The trouble with tracing. Midwest Studies in Philosophy 29(1):269–291. https://doi.org/10.1111/j.1475-4975.2005.00117.x

[86] Wallace E, Xiao K, Leike R, Weng L, Heidecke J, Beutel A (2024) The instruction hierarchy: training LLMs to prioritize privileged instructions. arXiv preprint arXiv:2404.13208, URL https://arxiv.org/abs/2404.13208

[87] Watson G (1996) Two faces of responsibility. Philosophical Topics 24(2):227–248. https://doi.org/10.5840/philtopics199624222

[88] Williams BAO (1965) Ethical consistency. Proceedings of the Aristotelian Society, Supplementary Volume 39:103–124. https://doi.org/10.1093/aristoteliansupp/39.1.103

[89] Willner Y, Heymann M (1991) Supervisory control of concurrent discrete-event systems. International Journal of Control 54(5):1143–1169. https://doi.org/10.1080/00207179108934202

[90] Wilsons & Clyde Coal Co Ltd v English (1938) [1938] AC 57 (HL); [1937] UKHL 2. House of Lords, judgment of 19 July 1937

[91] Wu Q, Bansal G, Zhang J, Wu Y, Li B, Zhu E, Jiang L, Zhang X, Zhang S, Liu J, Awadallah AH, White RW, Burger D, Wang C (2024) AutoGen: enabling next-gen LLM applications via multi-agent conversations. In: Conference on Language Modeling (COLM 2024), URL https://openreview.net/forum?id=BAakY1hNKS

[92] Yao ACC (1979) Some complexity questions related to distributive computing (preliminary report). In: Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing (STOC '79), pp 209–213, URL https://doi.org/10.1145/800135.804414

[93] Yao S, Zhao J, Yu D, Du N, Shafran I, Narasimhan K, Cao Y (2023) ReAct: synergizing reasoning and acting in language models. In: The Eleventh International Conference on Learning Representations (ICLR 2023), URL https://openreview.net/forum?id=WE_vluYUL-X

[94] Zimmerman MJ (1997) Moral responsibility and ignorance. Ethics 107(3):410–426. https://doi.org/10.1086/233742