

On the architecture of decomposable targets.

AlphaGeometry, healthcare payment, and the structure of hybrid reasoning.

abstract

The convolutional approach to learning problems treats a target variable as a single quantity to be approximated by a statistical model. We argue that a substantial class of important learning problems contains a deterministic component over a closed primitive set, and that for such problems the optimal architecture is decomposed: symbolic computation for the rule-governed component, statistical modeling for the residual. This architectural principle has been arrived at independently in two research conversations that have proceeded without significant contact — the neuro-symbolic program in artificial intelligence, exemplified by DeepMind’s AlphaGeometry, and the deterministic decomposition of healthcare payment developed in § M.01. We argue that the convergence is not coincidence. The two are instances of a general structural truth: when a target contains rule-governed structure, hybrid architectures dominate pure statistical approaches by a margin equal to the capacity reclaimed from approximating the rules. We name the principle, identify additional domains where it applies, and argue that the next significant capability advances in industrial analytics will come from recognizing decomposable structure in domains where the convolutional approach has been treated as adequate.

1 two conversations

In January 2024, DeepMind published a paper in *Nature* describing AlphaGeometry, a system capable of solving International Mathematical Olympiad geometry problems at near-gold-medal level. The paper attracted broad attention for its empirical result — solving twenty-five of thirty IMO geometry problems from the past quarter century, compared to ten by the previous state-of-the-art — and for the architecture that produced it. AlphaGeometry combines two distinct components. A symbolic deduction engine performs exact geometric reasoning by applying formal rules of inference to a problem’s premises, deriving conclusions through deterministic rule application. A neural language model, trained on one hundred million synthetic theorem-proof pairs, proposes auxiliary geometric constructions when the symbolic engine reaches an impasse. Neither component alone solves the problems. The symbolic engine cannot make the creative leaps required to introduce new geometric objects; the neural model cannot perform the rigorous step-by-step derivations required for valid proof. The combination produces what neither component could alone.¹

In April 2026, we published § M.01, *On the deterministic decomposition of healthcare payment*. The argument was that healthcare reimbursement, which the industry has treated as a single quantity to be forecasted, is in fact the sum of two epistemically distinct components: a deterministic contractual quantity C , computable exactly from the controlling rule system, and a probabilistic behavioral residual B , modelable by standard statistical methods once C is removed. The methodological prescription that follows is decomposed: compute C symbolically, model B statistically. The two

[1] marginalia

Trinh, T.H., Wu, Y., Le, Q.V. et al. Solving olympiad geometry without human demonstrations. Nature 625, 476–482 (2024). The architectural pattern was anticipated in earlier work but AlphaGeometry’s empirical demonstration was the inflection point at which the principle became visible to a general audience.

outputs serve different stakeholders and admit incompatible methodologies, and any approach that conflates them suffers a structural accuracy ceiling.

These two papers describe the same architectural principle. AlphaGeometry’s symbolic deduction engine and Crescent’s contractual computation engine are the same kind of component, performing the same kind of work: exact derivation from a closed primitive set. AlphaGeometry’s neural construction proposer and Crescent’s behavioral residual model are the same kind of component, performing the same kind of work: statistical inference over residual variance that the rule system cannot resolve. The two systems differ in domain (IMO geometry, healthcare claims), in symbolic apparatus (geometric inference, contract logic), and in the nature of the residual (creative construction, payer behavior). They are identical in architecture.

This is not coincidence. The two papers are instances of a general architectural principle that the convolutional approach to learning has been failing to recognize. We name the principle here, articulate why it works, and identify additional domains where it applies.

2 the principle

A learning problem is *decomposable* if its target variable Y can be written as $Y = D + S$, where D is computable from a closed primitive set by deterministic rule application, and S is the residual variance not derivable from rules. We have argued in § M.01 that decomposable problems admit a dominance result: any bounded statistical learner trained on Y is dominated by the same learner trained on S with D subtracted a priori. The dominance has two sources. First, on the rule-governed component, direct computation is exact while statistical approximation is bounded by the learner’s capacity. Second, on the residual, the learner trained on S allocates its full capacity to S , while the learner trained on Y must split its capacity between approximating D and modeling S .

The principle is general. It applies to any learning problem with a decomposable target. The question is which problems have this structure, and the answer turns out to be broader than the existing methodological practice would suggest. Geometric theorem proving is decomposable: D is the set of derivations from premises by formal inference rules, S is the choice of auxiliary constructions that the rule system does not determine but that creative search must propose. Healthcare reimbursement is decomposable: D is the contractually owed amount under the rule system, S is the payer’s behavioral deviation from contract. Tax computation is decomposable: D is the statutorily owed amount under the tax code, S is the residual associated with discretionary interpretations and enforcement behavior. Regulatory compliance scoring is decomposable: D is the determination of compliance under the regulatory framework, S is the variance in enforcement and interpretation. Contract pricing in regulated commodity derivatives is decomposable: D is the value derived from the contract’s deterministic payoff structure, S is the residual associated with counterparty behavior and market microstructure.

In each of these domains, the convolutional approach — model Y directly with a statistical learner — has been treated as the natural methodology. In each of these domains, the convolutional approach is dominated by the decomposed alternative. The dominance is not domain-specific; it follows from the structural properties of the target variable.

What has prevented recognition of this pattern? The answer is that the two communities working on it have been separated by disciplinary boundaries. The neuro-symbolic research program in AI has been working on the architectural principle in the context of mathematics, scientific reasoning, formal verification, and program synthesis — domains where the rule-governed component is itself a research object. The industrial analytics community has been working on the convolutional approximation of decomposable targets in domains like reimbursement, tax, compliance, and regulated derivatives — domains where the rule-governed component is a regulatory or contractual artifact rather than an object of research. Neither community has had reason to look at the other, and consequently the same architectural insight has been arrived at twice in parallel without unification.

The unification is the contribution of this note.

3 the neuro-symbolic program

The artificial intelligence research community has spent much of the past decade re-examining a question that was settled, prematurely, in the 1990s: whether intelligent systems should be built primarily on symbolic foundations (logic, formal rules, explicit representation) or on neural foundations (statistical learning, distributed representation, end-to-end optimization). The connectionist victory of the 2010s — large neural networks dominating benchmark after benchmark — appeared at the time to settle the question in favor of pure neural approaches. The settlement has been undone over the past five years.

The undoing has come from a specific pattern of empirical results. Pure neural systems excel at perceptual tasks (image classification, speech recognition), at fluent generation (language modeling, image synthesis), and at problems where the underlying structure can be learned from large datasets without explicit articulation. Pure neural systems struggle, sometimes catastrophically, at tasks that require exact rule application, multi-step compositional reasoning, generalization from few examples to novel rule combinations, and verifiable correctness. The struggles are not artifacts of insufficient scale; recent work has demonstrated that even very large neural models exhibit characteristic failure modes on rule-governed problems that scaling alone does not fix.

The neuro-symbolic response is to reintroduce explicit symbolic structure into the architecture, while retaining the neural component for what it does well. AlphaGeometry is one canonical example. DeepMind’s AlphaProof, released in 2024, is another — combining a symbolic theorem prover (Lean, the formal verification system used by professional mathematicians) with a neural model trained to propose proof tactics. AlphaProof achieved silver-medal performance on the 2024 IMO by deriving valid formal proofs through hybrid symbolic-neural search. The architecture

is the same as AlphaGeometry’s: symbolic engine handles exact derivation, neural model handles search guidance, neither component is sufficient alone.²

The pattern extends beyond mathematics. Neuro-symbolic concept learners combine neural perception with symbolic compositional reasoning to solve visual question-answering problems that pure neural approaches plateau on. Program synthesis systems combine type-theoretic constraints (symbolic) with neural search over candidate programs to generate code that satisfies formal specifications. Formal verification work combines SMT solvers (symbolic) with neural heuristics for proof search, producing verified software at scales previously infeasible. In each case the architectural pattern is the same: identify the component of the problem that is rule-governed, handle it with a symbolic system that guarantees correctness, and use a neural system for the residual that requires statistical pattern recognition rather than rule application.

The neuro-symbolic program has not yet been articulated as a general architectural principle — the literature treats it as a methodological tendency rather than a structural claim. But the principle is implicit in the empirical pattern. The systems that have achieved the most striking capability advances over the past three years have been hybrid architectures applied to problems with decomposable targets. The systems that have plateaued have been pure neural architectures applied to problems with rule-governed structure that the neural model is being asked to approximate.

4 healthcare payment is the same kind of problem

The argument of § M.01 was that healthcare payment is decomposable, that the industry has been modeling it convolutionally, and that the convolutional approach is dominated by the decomposed alternative. We now observe that healthcare payment is an instance of the same structural pattern that AlphaGeometry and AlphaProof have made visible in mathematics.

The structural correspondence is exact. AlphaGeometry’s symbolic deduction engine operates over the primitive set of geometric inference rules: points, lines, circles, angles, the axioms of Euclidean geometry, the rules of derivation that license one geometric fact from another. Crescent’s contractual computation engine operates over the primitive set of healthcare payment rules: the Code of Federal Regulations, the Medicare Physician Fee Schedule, the NCCI edits, the payer contract terms, the rules of derivation that license one payment fact from another. Both primitive sets are closed and finite. Both rule systems are functions from input state to output value. Both admit exact computation of the rule-governed component.

AlphaGeometry’s neural construction proposer learns from training data what kinds of auxiliary geometric constructions tend to enable proof completion. It does not derive the constructions; it proposes them. The constructions are then verified or rejected by the symbolic engine. Crescent’s behavioral residual model learns from training data what kinds of deviations from contractual payment a given payer tends to produce in given contexts. It does not derive the deviations from rules; it characterizes them statistically. The deviations are then incorporated into the cash forecast alongside the exact contractual computation.

[2] marginalia

DeepMind’s announcement of AlphaProof’s IMO 2024 performance is at deepmind.google/discover/blog/ai-solves-imo-problems-at-silver-medal-level. The formal write-up is forthcoming as of this note.

The two systems share the property that the symbolic component provides what the neural component cannot — exactness, verifiability, generalization from rules — while the neural component provides what the symbolic component cannot — pattern recognition over residuals that the rule system does not determine. Neither system would be improved by attempting to subsume one component into the other. AlphaGeometry would not be improved by replacing its symbolic engine with a larger neural model trained to approximate geometric inference. Crescent’s payment apparatus would not be improved by replacing its contractual computation with a larger neural model trained to approximate the payment rules. In both cases the hybrid architecture is the architecture that the structure of the problem demands.

This correspondence has practical consequences for how the work proceeds. Several follow.

First, the boundary between the symbolic and neural components is itself an object of design rather than a fixed property of the problem. AlphaGeometry’s authors made specific choices about which geometric facts the symbolic engine would derive automatically and which the neural model would need to propose. Crescent makes analogous choices about which payer behaviors are sub-deterministic enough to encode as near-rules in the symbolic apparatus and which are stochastic enough to model statistically. The boundary is sensitive to the resolution of the rule encoding and to the available training data for the neural component, and it shifts over time as both improve.

Second, the symbolic component carries the burden of correctness. In AlphaGeometry, a proof produced by the system is valid because each step in the derivation has been verified by the symbolic engine; the neural model’s role is to suggest steps, not to authorize them. In Crescent, the contractual computation is correct because each rule application has been verified against the controlling regulation or contract; the behavioral model’s role is to forecast deviation, not to determine entitlement. This asymmetric burden — symbolic for correctness, neural for search and approximation — is a stable property of well-designed hybrid systems.

Third, the neural component should be evaluated against the residual signal, not against the original target. AlphaGeometry’s neural model is not evaluated against geometric proof correctness directly; it is evaluated against its ability to propose constructions that, when added to the symbolic search space, enable proof completion. Crescent’s behavioral model is not evaluated against realized payment directly; it is evaluated against the residual $B = R - C$, which is the signal it is responsible for. Evaluating a neural component against the full target rather than against its residual responsibility is a category error that the convolutional approach commits routinely.

5 why the convergence has been slow

The neuro-symbolic research program and the decomposition of healthcare payment have arrived at the same architectural principle from different directions. Why has the convergence taken so long?

The answer is that the two communities have not been reading each other. Neuro-symbolic AI researchers work on problems that are recognized as AI problems: theorem proving, scientific reasoning, formal verification, program synthesis, visual reasoning. The domains are populated by people with AI research backgrounds. Healthcare reimbursement analytics, by contrast, is populated by people with backgrounds in actuarial science, operations research, and healthcare administration. The two communities attend different conferences, read different journals, and use different vocabularies. The same architectural insight, articulated in one community, does not transmit to the other because the channels do not exist.

The vocabulary problem is itself significant. The AI community talks about symbolic versus neural, about system 1 versus system 2, about explicit versus implicit representation. The healthcare analytics community talks about rules engines versus machine learning, about deterministic versus probabilistic, about deductive versus inductive methods. The terms map onto each other if you know they do, but the mapping is not obvious to someone working in only one community. A healthcare RCM practitioner reading the AlphaGeometry paper would not immediately recognize that the same architectural pattern applies to claim adjudication. An AI researcher looking at healthcare reimbursement would not immediately recognize that the same structural problem is present.

Beyond vocabulary, there is a more fundamental cultural difference. AI research treats the rule-governed component of a problem as something to be carefully designed — the formal rules of geometric inference, the type system of a programming language, the axioms of a logical system. Industrial analytics treats the rule-governed component as a given — the existing regulatory framework, the existing contract terms, the existing tax code. The first community sees rules as objects of research; the second sees them as constraints to be respected. Both views are correct in their respective contexts, but the difference in stance obscures the structural similarity of the architectural problems each community faces.

This is also a story about who gets to ask interesting research questions. The AlphaGeometry team at DeepMind has the institutional position, the computational resources, and the disciplinary license to ask “what is the right architecture for theorem proving?” The healthcare analytics teams at the major RCM vendors do not have the same institutional position to ask “what is the right architecture for claim adjudication?” The question gets asked instead at the perimeter — by independent researchers, by domain practitioners with quantitative backgrounds, by firms whose primary work is not analytics but who develop analytics in service of other ends. The peripheral position is what makes the recognition possible, because the peripheral observer is not committed to either community’s defaults.

6 where else this applies

The architectural principle, once named, predicts which industrial analytics domains are currently mispositioned. The prediction is specific: any domain whose target variable contains a rule-governed component over a closed primitive set, and which is currently being modeled convolutionally, will be dominated by a decomposed ap-

proach. The dominance is structural and follows from the same dominance argument as the healthcare payment case.

We identify several domains where the prediction applies.

Tax computation and tax compliance. The federal tax code, supplemented by state and local codes, treasury regulations, and case law, forms a closed rule system over the primitive set of taxpayer financial events. The contractually owed tax is exactly computable from the rule system, in the same sense that C is exactly computable in healthcare payment. The behavioral residual — including discretionary interpretations, audit risk, and enforcement variance — is modelable statistically. The industry currently treats tax computation as a hybrid of rule application (by tax preparation software) and human judgment (by accountants and tax attorneys), but the analytical infrastructure that supports tax-related financial decisions — tax provision modeling, transfer pricing analysis, tax-loss harvesting optimization — frequently treats the combined quantity as a single forecasted variable. The decomposition has not been articulated as a methodological principle.

Regulatory compliance scoring across industries. Financial regulation, environmental compliance, pharmaceutical regulation, and securities regulation each generate rule systems that produce binary or graded compliance determinations for regulated entities. Industry practice frequently uses statistical scoring models — machine learning classifiers trained on historical compliance outcomes — to predict compliance status. The compliance determination, however, is computable from the rules; what the statistical model is implicitly doing is approximating the rule system, badly, while the residual variance attributable to enforcement and interpretation goes unmodeled.

Pricing of regulated derivatives. Some derivatives have payoff structures that are exactly determined by contract terms and observable market variables — the deterministic component — while others have payoff structures that depend on counterparty behavior, settlement procedures, and market microstructure — the residual component. The industry has developed sophisticated pricing models for the determinate components (Black-Scholes and its extensions) but the residual components are frequently modeled by the same techniques, which is structurally inappropriate. The convolutional approach is more visible in certain over-the-counter derivatives markets where settlement behavior is itself a significant component of realized value.

Construction cost estimation under fixed-price contracts. Construction projects under fixed-price contracts have a rule-governed component (the contractually specified scope and unit prices) and a residual component (change orders, schedule variance, contractor behavior). Industry practice models total project cost convolutionally; the rule-governed component is computable from the contract and the residual is modelable statistically.

Compensation modeling in commission-based industries. Sales commissions, brokerage payments, and certain executive compensation arrangements are governed by contract terms that determine the rule-governed component exactly, with residual variance attributable to interpretation, exception handling, and discretionary adjustment. The convolutional approach treats expected compensation as a single forecasted quantity.

These domains share the structural property that justifies the architectural prediction. We do not claim that decomposed apparatus is the highest-priority work in each of these domains — that depends on commercial considerations specific to each — but we observe that the methodological mispositioning is present in each, and that the same architectural correction is available.³

7 what this means for the work

The unification of the neuro-symbolic AI research program and the deterministic decomposition of healthcare payment is not merely an intellectual observation. It has specific consequences for how the work proceeds.

The first consequence is that the apparatus being built at Crescent is not a health-care-specific rules engine. It is an instance of a general architectural pattern, and the design choices that govern its construction — boundary placement between symbolic and neural components, evaluation of components against their respective responsibilities, asymmetric correctness burdens — are the same design choices that govern hybrid AI systems generally. The literature on neuro-symbolic architectures is therefore directly relevant to the work, and the principles articulated in that literature inform the engineering decisions at every level.

The second consequence is that the residual component, B , deserves the kind of methodological care that the neuro-symbolic literature has developed for the neural components of hybrid systems. Specifically: B should be modeled with architectures that are designed for the structure of its variance, not with the convolutional methods that the industry has inherited from actuarial science. The behavioral residual is not a forecasting problem in the same sense that insurance loss forecasting is a forecasting problem. It is a residual variance problem, which has its own methodological literature — anomaly detection, residual learning, mixture models for sub-deterministic structure. The next phase of work on the behavioral component will draw on this literature rather than on the methodology that produced the seventy-to-eighty-percent ceiling we have been arguing against.

The third consequence is the most important, and we develop it at length. The boundary between the symbolic and neural components is not merely a research object — it is the locus of a self-improving loop that constitutes the actual compounding value of the apparatus over time.

The mechanism operates as follows. The neural component, modeling the behavioral residual B , identifies patterns in the residual that are stable across observations. Some of these patterns are genuinely stochastic — individual adjudicator discretion, timing artifacts, irreducible idiosyncrasy. Other patterns are sub-deterministic — internal payer policies that are not contained in the formal rule system but that are nonetheless consistent enough across cases to be characterized as near-rules. The neural component, in the course of modeling B , distinguishes between these two kinds of structure. The sub-deterministic patterns, once identified with sufficient stability, can be graduated into the symbolic apparatus. They cease to be neural-modeled residual and become symbolic-encoded near-rules. The residual that the neural component is

[3] marginalia

We do not enumerate all such domains here. The pattern, once seen, is easy to identify: look for a target variable that is being modeled convolutionally despite containing a rule-governed component derivable from a closed primitive set.

responsible for shrinks. The neural component’s capacity is reallocated to the new, smaller residual. The cycle continues.⁴

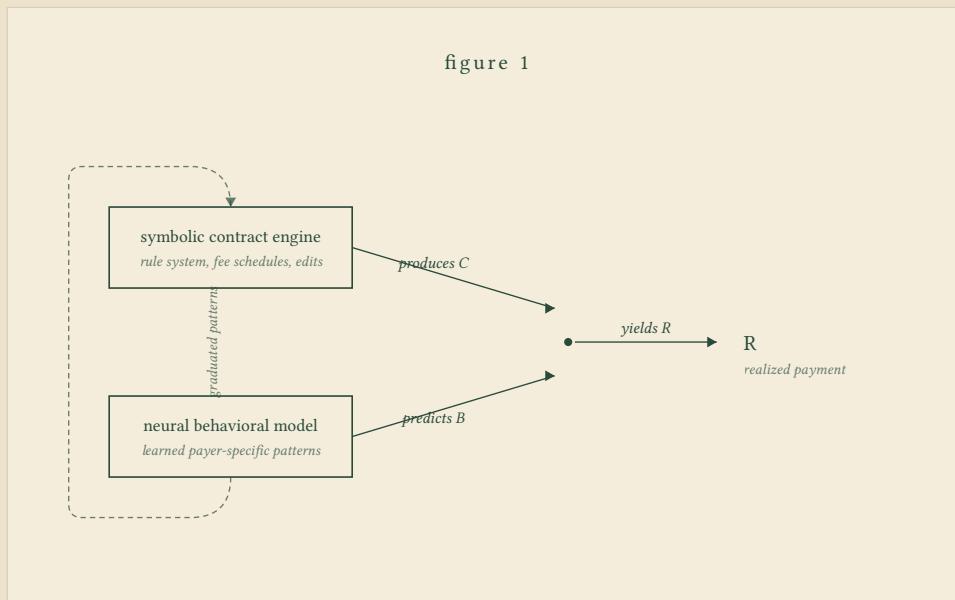
The correspondence to AlphaGeometry is exact. When AlphaGeometry’s neural model proposes an auxiliary construction that enables proof completion, the successful construction becomes part of the training data for subsequent runs. The neural model learns to propose better constructions; the symbolic engine receives improved suggestions; the system’s overall capability improves not through component scaling but through boundary migration. The boundary between what the symbolic engine derives automatically and what the neural model proposes is not fixed. It moves inward over time as more of what was previously creative-search becomes encoded as deductive pattern.

In the Crescent apparatus, the loop has a specific institutional manifestation. The behavioral model identifies that Payer X downcodes CPT 99214 to 99213 on Fridays in Q4 with eighty-three percent consistency. The pattern is stable enough to be characterized as a near-rule rather than as stochastic noise. The pattern is encoded into the symbolic apparatus as a payer-specific behavioral rule, distinguished from the contractual rules but treated with the same exactness of application. The residual that the neural model must characterize shrinks; the apparatus’s overall accuracy improves; the institutional knowledge of payer behavior accumulates not in human heads but in encoded rules.

This mechanism is the source of the apparatus’s compounding value. A static rules engine is a snapshot of the rule system as encoded at one moment. A static neural

[4] marginalia

The loop mechanism is not unique to neuro-symbolic systems. It is the basic structure of any system in which an explicit and an implicit representation coexist and exchange material over time. The neuro-symbolic literature has articulated it most clearly because the two representations are most clearly separated.



the architecture of decomposable hybrid reasoning. The symbolic contract engine produces C (the calculable component) from the rule system. The neural behavioral model predicts B (the residual) from learned payer-specific patterns. Their sum yields R (the realized payment). The dashed feedback path represents the graduation mechanism: patterns that stabilize in the neural component over time become candidates for absorption into the symbolic component as rules. The loop monotonically reduces the share of variance carried by the neural component as the symbolic component grows.

model is a snapshot of payer behavior as observed in one training window. Both depreciate as the world changes — payers update their policies, contracts get renegotiated, regulations are amended. A looping apparatus does not depreciate in the same way. As the world changes, the neural component identifies the new patterns; the patterns that prove stable get graduated to the symbolic component; the apparatus adapts to its environment through the same mechanism that produces its initial capability. The apparatus's institutional knowledge accumulates over time in a way that no static system can match, and the accumulation is the moat.

The loop also has an epistemological implication that deserves explicit articulation. The line between epistemic uncertainty (uncertainty due to missing information) and aleatory uncertainty (irreducible randomness) is not fixed. It depends on how much information has been extracted from the system. What appears genuinely stochastic at one stage of the loop may reveal itself to be sub-deterministic at the next stage, as more patterns get identified and encoded. A well-designed loop apparatus moves this line over time, converting epistemic uncertainty into encoded knowledge and revealing the true aleatory floor. The asymptotic goal of the apparatus is not the perfect prediction of B ; it is the perfect separation of B into its sub-deterministic and genuinely stochastic components, with the former encoded as symbolic near-rules and only the latter modeled statistically. The asymptote may not be reachable in practice, but it is the right asymptote.

The fourth consequence is that the work has natural extensibility beyond healthcare. The apparatus being built — the methodology for identifying rule-governed structure, the engineering for encoding closed primitive sets, the statistical machinery for residual modeling, and the loop mechanism for graduating sub-deterministic patterns into the symbolic component — generalizes. Each of the domains identified in section 6 is a potential application of the same architectural principle and the same engineering apparatus. We do not commit here to any specific extension, but we note that the structural similarity is real, and that the apparatus is designed with the generality in mind.

8 closing

We have argued that healthcare payment and AlphaGeometry are instances of the same architectural principle: that targets with rule-governed structure are best handled by hybrid systems that compute the rule-governed component symbolically and model the residual statistically. The principle has been arrived at independently in two research communities that have not been reading each other, and the convergence is evidence that the principle is general rather than domain-specific.

The convolutional approach — model the combined target with a statistical learner — is dominated for any decomposable problem. The dominance is mathematical, not empirical. It follows from the capacity-allocation argument in § M.01 and is reinforced by the empirical pattern of neuro-symbolic AI results. Domains that contain decomposable structure but are currently treated convolutionally are mispositioned, and the correction available is the same correction that AlphaGeometry demonstrated in mathematics and that § M.01 articulated for healthcare payment.

The work at Crescent Research proceeds from this position. The healthcare-specific apparatus is one application of a general architectural principle. The principle predicts the existence of a self-improving loop between the symbolic and neural components, and the loop is the source of the apparatus's compounding institutional value.

see also

§ M.01 On the deterministic decomposition of healthcare payment.