

Article

Not peer-reviewed version

MetaStructure and Iterated MetaStructure: MetaCognition, MetaLaerning, MetaAnalysis, MetaScience, MetaPhilosophy, and More

[Takaaki Fujita](#)*

Posted Date: 27 August 2025

doi: 10.20944/preprints202508.1925.v1

Keywords: MetaStructure; Iterated MetaStructure; MetaCognition; Meta-Laerning; Meta-Analysis;
MetaScience; Metaphilosophy



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

MetaStructure and Iterated MetaStructure: MetaCognition, MetaLearning, MetaAnalysis, MetaScience, MetaPhilosophy, and More

Takaaki Fujita

Independent Researcher, Tokyo, Japan; takaaki.fujita060@gmail.com

Abstract

A *MetaStructure* is a higher-level framework that treats entire collections of structures as single objects, equipped with natural operations that preserve isomorphisms across different domains. The term "Structure" here refers broadly to mathematical systems as well as real-world models. An *Iterated MetaStructure* generalizes this idea recursively, generating successive layers in which structures of structures form deeper hierarchical meta-levels. In this paper, we investigate whether well-known meta-concepts such as MetaCognition, Meta-Learning, Meta-Analysis, MetaScience, and Metaphilosophy can be extended by means of Iterated MetaStructures. While some of these meta-concepts are primarily conceptual in nature and lack precise mathematical definitions, we deliberately attempt to formalize them as rigorously as possible and examine whether such formalizations can be systematically extended.

Keywords: MetaStructure; Iterated MetaStructure; MetaCognition; Meta-Learning; Meta-Analysis; MetaScience; Metaphilosophy

1. Preliminaries

This section presents the fundamental concepts and definitions that underpin the discussions in this paper. Throughout this paper, all structures and sets are assumed to be finite.

1.1. Classical Structure

In this paper, the term *Structure* refers broadly to a mathematical system, not restricted to a single area, but encompassing domains such as Set Theory, Logic, Probability, Statistics, Algebra, and Geometry.

Definition 1.1 (Classical Structure). (cf.[1,2]) A *Classical Structure* C is a mathematical object arising from a traditional field—for example Set Theory, Logic, Probability, Statistics, Algebra, Geometry, Graph Theory, Automata Theory, or Game Theory. Formally, it may be represented as a pair

$$C = (H, \{\#^{(m)}\}_{m \in \mathcal{I}}),$$

where:

- H is a nonempty set, often called the *carrier* or *universe*.
- For each $m \in \mathcal{I} \subseteq \mathbb{Z}_{>0}$, there exists an m -ary operation

$$\#^{(m)} : H^m \longrightarrow H,$$

subject to appropriate *axioms* (such as associativity, commutativity, or identity laws), which vary according to the chosen type of structure.

The collection $\{\#^{(m)} : m \in \mathcal{I}\}$ determines the *type* of C . Representative examples include:

- A *Set* (S, \emptyset) , consisting solely of a carrier with distinguished elements or relations, but without operations [3,4].
- A *Logic* structure (L, \wedge, \vee, \neg) , where \wedge, \vee are binary connectives and \neg a unary connective, satisfying logical axioms [5].
- A *Probability* model (Ω, \mathcal{F}, P) , where $P : \mathcal{F} \rightarrow [0, 1]$ is a probability measure on a sigma-algebra $\mathcal{F} \subseteq \mathcal{P}(\Omega)$ [6,7].
- A *Statistical* model (X, \mathcal{A}, θ) , where θ maps data X into parameters of interest [8,9].
- *Algebraic* structures such as:
 - A *Group* $(G, *)$, with $* : G \times G \rightarrow G$ satisfying associativity, identity, and inverses [10,11].
 - A *Ring* $(R, +, \times)$, with two binary operations fulfilling ring axioms [12,13].
 - A *Vector Space* $(V, +, \cdot)$ over a field \mathbb{F} , with scalar multiplication $\cdot : \mathbb{F} \times V \rightarrow V$ [14–16].
- A *Geometric* structure (X, dist) , where $\text{dist} : X \times X \rightarrow \mathbb{R}$ satisfies the axioms of a metric.
- A *Graph* (V, E) , where $E \subseteq \{\{u, v\} \mid u, v \in V\}$ for undirected graphs, or $E \subseteq V \times V$ for directed graphs, with adjacency and incidence relations [17–20].
- An *Automaton* $(Q, \Sigma, \delta, q_0, F)$, where Q is a set of states, Σ an input alphabet, $\delta : Q \times \Sigma \rightarrow Q$ the transition function, $q_0 \in Q$ the start state, and $F \subseteq Q$ the accepting states [21,22].
- A *Game* $(N, \{A_i\}, \{u_i\})$, where N is the set of players, A_i each player's action set, and $u_i : \prod_{j \in N} A_j \rightarrow \mathbb{R}$ the payoff function for player i [23,24].

Related concepts include the *HyperStructure* [25–28] and the *SuperHyperStructure* [29–32], which have also been extensively investigated in recent studies.

1.2. MetaStructure (Structure of a Structure)

Fix once and for all a single-sorted, finitary *signature*

$$\Sigma = (\text{Func}, \text{Rel}, \text{ar}_{\text{Func}}, \text{ar}_{\text{Rel}}),$$

where *Func* (resp. *Rel*) is a set of function (resp. relation) symbols and *ar* records their arities. A (single-sorted) Σ -*structure* is a tuple

$$\mathbf{C} = (H, (f^{\mathbf{C}})_{f \in \text{Func}}, (R^{\mathbf{C}})_{R \in \text{Rel}}),$$

consisting of a nonempty carrier H , together with interpretations $f^{\mathbf{C}} : H^m \rightarrow H$ for each $f \in \text{Func}$ of arity m , and relations $R^{\mathbf{C}} \subseteq H^r$ for each $R \in \text{Rel}$ of arity r . Let Str_{Σ} denote the class of all such structures.

Definition 1.2 (MetaStructure over a fixed signature). (cf. [33]) With Σ as above, a *MetaStructure* (a “structure of structures”) is a pair

$$\mathbb{M} = (U, (\Phi_{\ell})_{\ell \in \Lambda}),$$

where:

- $U \subseteq \text{Str}_{\Sigma}$ is a nonempty collection of Σ -structures (the level-0 objects);
- for each label $\ell \in \Lambda$ with *meta-arity* $k_{\ell} \in \mathbb{N}$, the *meta-operation*

$$\Phi_{\ell} : U^{k_{\ell}} \longrightarrow U$$

is described by uniform *constructors* acting on carriers and symbol interpretations:

$$\begin{aligned} \Gamma_{\ell}(\mathbf{C}_1, \dots, \mathbf{C}_{k_{\ell}}) &= H_{\ell} && \text{(functorially built carrier);} \\ \forall f \in \text{Func} : f^{\Phi_{\ell}(\mathbf{C}_1, \dots, \mathbf{C}_{k_{\ell}})} &= \Lambda_{\ell}^f(f^{\mathbf{C}_1}, \dots, f^{\mathbf{C}_{k_{\ell}}}); \\ \forall R \in \text{Rel} : R^{\Phi_{\ell}(\mathbf{C}_1, \dots, \mathbf{C}_{k_{\ell}})} &= \Xi_{\ell}^R(R^{\mathbf{C}_1}, \dots, R^{\mathbf{C}_{k_{\ell}}}), \end{aligned}$$

where the recipes Λ_{ℓ}^f and Ξ_{ℓ}^R depend only on f, R and ℓ (not on the particular representatives) and produce the output interpretations over H_{ℓ} .

Each Φ_ℓ is required to be *isomorphism-invariant* (natural): if $\alpha_i : \mathbf{C}_i \xrightarrow{\cong} \mathbf{D}_i$ are isomorphisms for $1 \leq i \leq k_\ell$, then there is an induced isomorphism

$$\Phi_\ell(\alpha_1, \dots, \alpha_{k_\ell}) : \Phi_\ell(\mathbf{C}_1, \dots, \mathbf{C}_{k_\ell}) \xrightarrow{\cong} \Phi_\ell(\mathbf{D}_1, \dots, \mathbf{D}_{k_\ell}),$$

compatible with all function and relation symbols of Σ .

1.3. Iterated MetaStructure (Structure of Structure of ... of Structure)

An *Iterated MetaStructure* is obtained by repeatedly applying the MetaStructure construction, thereby forming successive levels where “structures of structures” build a hierarchical tower (cf. [33–36]).

Definition 1.3 (Iterated MetaStructure of depth t). (cf. [33]) For $t \in \mathbb{N}$, an *Iterated MetaStructure of depth t* over Σ is a MetaStructure $\mathfrak{M}^{(t)}$ obtained by t iterations of a lifting procedure. When $s < t$, we *lift* a height- s MetaStructure $\mathfrak{M}^{(s)} = (U^{(s)}, \{\odot_i\}, \{\mathcal{S}_j\})$ to height t by

$$\iota_{s \rightarrow t} : U^{(s)} \xrightarrow{U_\Sigma^{t-s}} U^{(t)} := U_\Sigma^{t-s}(U^{(s)}),$$

and, for each meta-operation $\odot_i : (E_\Sigma^{m_i})^{k_i} \rightarrow \mathcal{P}^{n_i}(E_\Sigma^{n_i})$, define its lift by

$$\odot_i^\uparrow : (E_\Sigma^{m_i+t-s})^{k_i} \longrightarrow \mathcal{P}^{n_i}(E_\Sigma^{n_i+t-s}), \quad \odot_i^\uparrow(U_\Sigma^{t-s}(x_1), \dots, U_\Sigma^{t-s}(x_{k_i})) := U_\Sigma^{t-s}(\odot_i(x_1, \dots, x_{k_i})),$$

and analogously for relations $\mathcal{S}_j^\uparrow := (U_\Sigma^{t-s})^{\times \ell_j}(\mathcal{S}_j)$.

2. Main Results: Some Iterated MetaStructures

This section presents the main results of the paper.

2.1. Iterated-Metacognition (Cognition of ... of Cognitions)

Metacognition is cognition about cognition: monitoring, evaluating, and regulating one’s thinking, strategies, confidence, and control during problem solving and reflection (cf. [37–41]). Iterated-Metacognition recursively applies metacognition to itself, forming hierarchical layers that optimize monitoring, control, calibration, and strategy selection across tasks continually.

Definition 2.1 (Metacognition). Let a *cognitive learner* produce a hypothesis $h \in \mathcal{H}$ from data $D \in \mathcal{D}$ via $A : \mathcal{D} \rightarrow \mathcal{H}$. A *metacognitive policy* is a measurable map

$$\mathcal{M} : \mathcal{H} \times \mathcal{D} \longrightarrow \mathcal{H} \times [0, 1], \quad (h, D) \mapsto (h^+, c),$$

that (i) *monitors* by outputting a calibrated confidence c and (ii) *controls* by updating h to h^+ . Given a data-label distribution P on $\mathcal{X} \times \mathcal{Y}$ and loss ℓ , metacognition seeks

$$\mathcal{M}^* \in \arg \min_{\mathcal{M}} \mathbb{E}_{D \sim \mathcal{D}} \mathbb{E}_{(x,y) \sim P} \left[\ell(h^+(x), y) + \lambda (c - \mathbf{1}\{h^+(x) = y\})^2 \right],$$

with $\lambda \geq 0$ penalizing miscalibration.

Example 2.2 (Metacognition (everyday study)). A student preparing for a calculus exam actively regulates their own thinking:

- (i) **Monitor:** While reading, they notice: “I follow worked examples but cannot derive the theorem unaided.”
- (ii) **Evaluate:** They judge understanding as recognition-only and overconfident on integrals.
- (iii) **Control:** They switch to retrieval practice (closed-book problems), set spaced reviews, and keep an error log to target weak steps.

This is first-order metacognition: monitoring, evaluating, and controlling one’s ongoing cognition.

Definition 2.3 (Iterated–Metacognition (depth t)). Let a *cognitive learner* $A^{(0)} : \mathcal{D} \rightarrow \mathcal{H}$ map data $D \in \mathcal{D}$ to a hypothesis $h \in \mathcal{H}$. A level-1 *metacognitive policy* (cf. Definition 2.1) is a measurable map

$$\mathcal{M}^{(1)} : \mathcal{H} \times \mathcal{D} \longrightarrow \mathcal{H} \times [0, 1], \quad (h, D) \mapsto (h^+, c),$$

where h^+ is a controlled update of h and c is a calibrated confidence. Fix a base loss $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow [0, \infty)$ and constants $\lambda, \mu \geq 0$. For a data–label distribution P on $\mathcal{X} \times \mathcal{Y}$, the level-1 *metacognitive risk* is

$$\mathcal{R}^{(1)}(\mathcal{M}^{(1)}; P) := \mathbb{E}_{D \sim \mathcal{D}} \mathbb{E}_{(x,y) \sim P} \left[\underbrace{\ell(h^+(x), y)}_{\text{task loss}} + \lambda \underbrace{(c - \mathbf{1}\{h^+(x) = y\})^2}_{\text{calibration penalty}} + \mu \underbrace{\text{Cost}(h, h^+)}_{\text{control cost}} \right], \quad (1)$$

with a measurable control cost $\text{Cost} : \mathcal{H} \times \mathcal{H} \rightarrow [0, \infty)$.

For $t \geq 2$, define a level- $(t-1)$ *task* as an evaluation protocol $U_{t-1} = (\Pi_{t-2}, \text{protocol})$ specifying how to score level- $(t-1)$ policies via $\mathcal{R}^{(t-1)}$. Let \mathcal{T}_{t-1} be the space of such tasks and Π_{t-1} a distribution on \mathcal{T}_{t-1} . A level- t *meta-experience* is

$$\mathcal{E}^{(t)} = \{(U_{t-1}^{(j)}, \mathcal{D}^{(j)})\}_{j=1}^m, \quad U_{t-1}^{(j)} \stackrel{\text{i.i.d.}}{\sim} \Pi_{t-1},$$

where each $\mathcal{D}^{(j)}$ bundles the data required by $U_{t-1}^{(j)}$. A level- t *iterated metacognitive learner* outputs a level- $(t-1)$ policy:

$$\mathcal{M}^{(t)} : \bigcup_{m \geq 1} (\mathcal{T}_{t-1} \times \text{Data}_{t-1})^m \longrightarrow \{\mathcal{M}^{(t-1)}\}, \quad \mathcal{E}^{(t)} \mapsto \mathcal{M}_{\mathcal{E}^{(t)}}^{(t-1)}.$$

The level- t risk is defined *recursively* by

$$\mathcal{R}^{(t)}(\mathcal{M}^{(t)}; \Pi_{t-1}) := \mathbb{E}_{\mathcal{E}^{(t)} \sim \Pi_{t-1}} \mathbb{E}_{U_{t-1} \sim \Pi_{t-1}} \mathbb{E}_{\mathcal{D} \sim U_{t-1}} \mathcal{R}^{(t-1)}(\mathcal{M}^{(t)}(\mathcal{E}^{(t)}); U_{t-1}), \quad (2)$$

with the base case $\mathcal{R}^{(1)}$ given by (1). We call any $\mathcal{M}^{(t)}$ minimizing (2) an *Iterated–Metacognition* (depth t) solution.

Example 2.4 (Iterated–Metacognition (everyday refinement of the strategy)). The same student reflects on their *metacognitive method* a week later:

- (i) **Meta-monitor:** For each problem, they recorded confidence $c \in [0, 1]$ and correctness $\mathbf{1}\{\text{correct}\}$. They now compute average calibration error $\bar{e} = \frac{1}{n} \sum_{k=1}^n |c_k - \mathbf{1}\{\text{correct}_k\}|$.
- (ii) **Meta-evaluate:** They find overconfidence on word problems (\bar{e} is largest there) and that rereading yields poor retention versus self-testing.
- (iii) **Meta-control:** They revise the *metacognitive policy*: tighten the confidence threshold for moving on (require $c \geq 0.9$ with two consecutive correct recalls), prioritize interleaved retrieval over rereading, and schedule next-day “calibration checks” to update thresholds if $\bar{e} > 0.15$.

This is iterated–metacognition: applying metacognition to one’s own metacognitive strategy and updating the policy itself.

Definition 2.5 (A signature for metacognition by depth). Let $\Sigma_{\text{MC}}^{(0)}$ have carrier $H^{(0)} = \mathcal{H}$ and function symbols

$$\text{Apply}^{(0)} : \mathcal{H} \times \mathcal{X} \rightarrow \mathcal{Y}, \quad \text{Loss}^{(0)} : \mathcal{H} \times (\mathcal{X} \times \mathcal{Y}) \rightarrow \mathbb{R}_{\geq 0}.$$

For $t \geq 1$, extend to $\Sigma_{\text{MC}}^{(t)}$ with carrier $H^{(t)} = \text{Pol}^{(t-1)}$ (the set of level- $(t-1)$ metacognitive policies) and symbols

$$\text{Train}^{(t)} : \text{Exp}^{(t)} \rightarrow \text{Pol}^{(t-1)}, \quad \text{Risk}^{(t)}(\mathcal{M}^{(t-1)}, R) \text{ encoding } R = \mathcal{R}^{(t-1)}(\mathcal{M}^{(t-1)}; \cdot).$$

Proposition 2.6 (Meta-operations at depth t). For $\mathbf{C}_1^{(t)}, \dots, \mathbf{C}_k^{(t)} \Sigma_{\text{MC}}^{(t)}$ -structures with carriers $H_i^{(t)}$:

- (i) **Tagged sum** $\Phi_{\oplus}^{(t)}$: set $H_{\oplus}^{(t)} := \prod_{i=1}^k H_i^{(t)}$ and define $\text{Train}_{\oplus}^{(t)}(\mathcal{Z}^{(t)}, i) := \text{Train}_{C_i}^{(t)}(\mathcal{Z}^{(t)})$; interpret $\text{Risk}_{\oplus}^{(t)}$ componentwise on tagged inputs.
- (ii) **Weighted ensemble (shared protocol)** $\Phi_{\text{ens}}^{(t)}$: if the evaluation protocol agrees, set $H_{\text{ens}}^{(t)} := (\prod_{i=1}^k H_i^{(t)}) \times \Delta^{k-1}$ and

$$\text{Train}_{\text{ens}}^{(t)}(\mathcal{Z}^{(t)}) = \left(\text{Train}_{C_1}^{(t)}(\mathcal{Z}^{(t)}), \dots, \text{Train}_{C_k}^{(t)}(\mathcal{Z}^{(t)}), \hat{w}(\mathcal{Z}^{(t)}) \right),$$

where $\hat{w} \in \Delta^{k-1}$ minimizes an empirical proxy of $\mathcal{R}^{(t-1)}$.

Theorem 2.7 (Iterated–Metacognition generalizes Metacognition and is an Iterated MetaStructure).

For every $t \geq 1$:

- (a) **(Generalization)** Fix a level- $(t-1)$ task $U_{t-1}^* \in \mathcal{T}_{t-1}$ and take $\Pi_{t-1} := \delta_{U_{t-1}^*}$. If $\mathcal{M}^{(t)}$ ignores $\mathcal{Z}^{(t)}$ and always returns a fixed level- $(t-1)$ policy $\mathcal{B}^{(t-1)}$, then

$$\mathcal{R}^{(t)}(\mathcal{M}^{(t)}; \Pi_{t-1}) = \mathbb{E}_{\mathcal{D}' \sim U_{t-1}^*} \mathcal{R}^{(t-1)}(\mathcal{B}^{(t-1)}; U_{t-1}^*).$$

In particular, when $t = 1$ and $U_0^* = (P, \ell, \lambda, \mu)$, Iterated–Metacognition reduces to the (single-level) Metacognition risk (1).

- (b) **(Iterated MetaStructure)** For each t , with $U^{(t)}$ the class of $\Sigma_{\text{MC}}^{(t)}$ –structures (Definition 2.5) and meta-operations $\{\Phi_{\oplus}^{(t)}, \Phi_{\text{ens}}^{(t)}\}$ (Proposition 2.6),

$$\mathfrak{M}^{(t)} := (U^{(t)}, \{\Phi_{\oplus}^{(t)}, \Phi_{\text{ens}}^{(t)}\})$$

is a MetaStructure over $\Sigma_{\text{MC}}^{(t)}$ in the sense of Definition 1.2. Moreover, if $s < t$, the lifted pair $\mathfrak{M}^{(s)\uparrow}$ (Definition 1.3) is again a MetaStructure on $U^{(t)}$.

Proof. (a) *Generalization.* By (2) and $\Pi_{t-1} = \delta_{U_{t-1}^*}$,

$$\begin{aligned} \mathcal{R}^{(t)}(\mathcal{M}^{(t)}; \Pi_{t-1}) &= \mathbb{E}_{\mathcal{Z}^{(t)} \sim \delta_{U_{t-1}^*}} \mathbb{E}_{U_{t-1} \sim \delta_{U_{t-1}^*}} \mathbb{E}_{\mathcal{D}' \sim U_{t-1}^*} \mathcal{R}^{(t-1)}(\mathcal{M}^{(t)}(\mathcal{Z}^{(t)}); U_{t-1}^*) \\ &= \mathbb{E}_{\mathcal{D}' \sim U_{t-1}^*} \mathcal{R}^{(t-1)}(\underbrace{\mathcal{M}^{(t)}(\mathcal{Z}^{(t)}); U_{t-1}^*}_{\equiv \mathcal{B}^{(t-1)}}) \\ &= \mathbb{E}_{\mathcal{D}' \sim U_{t-1}^*} \mathcal{R}^{(t-1)}(\mathcal{B}^{(t-1)}; U_{t-1}^*). \end{aligned}$$

When $t = 1$, $U_0^* = (P, \ell, \lambda, \mu)$ and $\mathcal{R}^{(0)}$ collapses to task loss, so (1) is recovered. For concreteness, if $\lambda = 1$, $\mu = 0$, and for some (x, y) the adapted leaf predicts correctly with $c = 0.8$, then the per-sample penalty equals $(0.8 - 1)^2 = 0.04$; if it errs with $c = 0.2$, the penalty $(0.2 - 0)^2 = 0.04$ is symmetric, illustrating calibrated control inside (1).

(b) *Iterated MetaStructure.* Fix $t \geq 1$. Uniform carrier constructors. By Definition 2.5, each structure $\mathbf{C}^{(t)}$ has carrier $H^{(t)} = \text{Pol}^{(t-1)}$. Proposition 2.6 specifies for any finite family $(C_i^{(t)})_{i=1}^k$ the carriers $H_{\oplus}^{(t)} = \prod_i H_i^{(t)}$ and $H_{\text{ens}}^{(t)} = (\prod_i H_i^{(t)}) \times \Delta^{k-1}$, which depend only on the inputs' carriers, as required by Definition 1.2.

Uniform symbol interpretations. The recipes for $\text{Train}^{(t)}$ and $\text{Risk}^{(t)}$ under $\Phi_{\oplus}^{(t)}$ and $\Phi_{\text{ens}}^{(t)}$ (tag selection and convex weighting of policies) are independent of representatives, hence define Λ_{ℓ}^f and Ξ_{ℓ}^R uniformly for all inputs (cf. Definition 1.2).

Isomorphism invariance (naturality). Let $\alpha_i : C_i^{(t)} \rightarrow D_i^{(t)}$ be $\Sigma_{\text{MC}}^{(t)}$ –isomorphisms, i.e. bijections $\alpha_i : H_i^{(t)} \rightarrow H_i^{(t)}$ intertwining $\text{Train}^{(t)}$ and preserving $\text{Risk}^{(t)}$:

$$\text{Risk}_{C_i}^{(t)}(\mathcal{M}, R) \iff \text{Risk}_{D_i}^{(t)}(\alpha_i(\mathcal{M}), R).$$

Define $\alpha_{\oplus}^{(t)} := \prod_i \alpha_i$ and $\alpha_{\text{ens}}^{(t)} := (\prod_i \alpha_i) \times \text{id}_{\Delta^{k-1}}$. Then, for all $\mathcal{Z}^{(t)}$,

$$\text{Train}_{D_{\oplus}}^{(t)}(\alpha_{\oplus}^{(t)}(\cdot), \mathcal{Z}^{(t)}) = \alpha_{\oplus}^{(t)}\left(\text{Train}_{C_{\oplus}}^{(t)}(\cdot, \mathcal{Z}^{(t)})\right), \quad \text{and} \quad \text{Risk}^{(t)} \text{ is preserved by the same bijections.}$$

Hence there are induced isomorphisms $\Phi_{\oplus}^{(t)}(\alpha_1, \dots, \alpha_k)$ and $\Phi_{\text{ens}}^{(t)}(\alpha_1, \dots, \alpha_k)$, verifying naturality. Therefore $\mathfrak{M}^{(t)}$ satisfies the axioms of a MetaStructure (Definition 1.2).

Lifted structures. If $s < t$, the lift $\iota_{s \rightarrow t}$ (Definition 1.3) acts symbolwise, replacing each depth- s symbol by its depth- t analogue and composing meta-operations componentwise. Typing and naturality are preserved, so $\mathfrak{M}^{(s)\uparrow}$ is a MetaStructure on $U^{(t)}$.

Combining the three underlined parts yields (b). \square

2.2. Iterated-Meta-Learning (Learning of ... of Learnings)

Learning updates a model or behavior from data and feedback, minimizing loss, improving predictions, skills, or decisions over time continually. Meta-Learning trains across tasks to learn learning strategies, initializations, or update rules enabling rapid adaptation, few-shot generalization, and robustness, transferability (cf.[42,43]). Iterated-Meta-Learning composes multiple meta-levels, where meta-learners optimize other meta-learners, yielding recursive improvement, self-referential curricula, continual adaptation, and automation capabilities emergence.

Definition 2.8 (Base Learning (level 0)). A task is $T = (\mathcal{X}_T, \mathcal{Y}_T, P_T, \mathcal{H}_T, \ell_T)$, where P_T is a distribution on $\mathcal{X}_T \times \mathcal{Y}_T$ and $\ell_T : \mathcal{Y}_T \times \mathcal{Y}_T \rightarrow [0, \infty)$. A level-0 learner $A^{(0)}$ maps a sample $S \in (\mathcal{X}_T \times \mathcal{Y}_T)^*$ to a hypothesis $h_S \in \mathcal{H}_T$. Its *true risk* on T is

$$\mathcal{L}^{(0)}(A^{(0)}; T) := \mathbb{E}_{S \sim P_T^n} \mathbb{E}_{(x,y) \sim P_T} \ell_T(h_S(x), y).$$

Example 2.9 (Learning (level 0, everyday life)). A home cook learns to bake a single sourdough loaf.

- (i) **Task T .** Input \mathcal{X}_T : flour type, hydration, proof time, oven temperature. Output \mathcal{Y}_T : loaf quality score (crumb, rise, taste). Loss ℓ_T : deviation from target score.
- (ii) **Data S .** They run a few attempts with different hydrations and proof times, logging outcomes.
- (iii) **Hypothesis h_S .** A concrete recipe (e.g., 72% hydration, 24h cold proof, bake at 240°C with steam).
- (iv) **Learning.** The cook updates h_S to reduce ℓ_T on subsequent bakes (adjusts hydration and proof until target quality is met).

This is base learning $A^{(0)} : S \mapsto h_S$ for a *single* task T .

Definition 2.10 (Meta-Learning (level 1)). Let Π_0 be a distribution over level-0 tasks T . A level-1 learner $A^{(1)}$ maps a *meta-experience*

$$\mathcal{E}^{(1)} = \{(T^{(j)}, S^{(j)})\}_{j=1}^m, \quad T^{(j)} \stackrel{\text{i.i.d.}}{\sim} \Pi_0, \quad S^{(j)} \sim P_{T^{(j)}}^{n_j},$$

to a base learner $A_{\mathcal{E}^{(1)}}^{(0)}$. Its *meta-risk* is

$$\mathcal{L}^{(1)}(A^{(1)}; \Pi_0) := \mathbb{E}_{\mathcal{E}^{(1)} \sim \Pi_0} \mathbb{E}_{T \sim \Pi_0} \mathbb{E}_{S' \sim P_{T'}^{n'}} \mathcal{L}^{(0)}(A_{\mathcal{E}^{(1)}}^{(0)}; T).$$

Example 2.11 (Meta-Learning (level 1, everyday life)). A caterer must quickly learn *new* recipes across cuisines before events.

- (i) **Task distribution Π_0 .** Each task $T^{(j)}$ is a different dish (Thai curry, gluten-free cake, vegan stew), each with its own inputs/outputs and loss.
- (ii) **Meta-experience $\mathcal{E}^{(1)}$.** For many past dishes, the caterer has small practice sets $S^{(j)}$ (pilot batches) and observed errors.
- (iii) **Output (a learner).** From $\mathcal{E}^{(1)}$, they learn a *recipe-bootstrapping procedure* $A_{\mathcal{E}^{(1)}}^{(0)}$:
 - start with a “base” template (mise en place checklist, temperature ladder),
 - run a two-point pilot (low/high spice or hydration),
 - update the working recipe by one-step gradient: increase/decrease limiting factor causing largest error,

- finalize after a calibration taste test.

(iv) **Adaptation to a new dish.** Given a *new* task $T \sim \Pi_0$, applying $A_{\mathcal{G}^{(1)}}^{(0)}$ yields a good recipe after very few trials (few-shot adaptation).

This is level-1 meta-learning $A^{(1)} : \mathcal{G}^{(1)} \mapsto A_{\mathcal{G}^{(1)}}^{(0)}$ minimizing $\mathcal{L}^{(1)}$ over Π_0 .

Definition 2.12 (Iterated–Meta–Learning (level $t \geq 2$)). For $t \geq 2$, let a level- $(t-1)$ task be a protocol $U_{t-1} = (\Pi_{t-2}, \text{eval})$ specifying how to evaluate level- $(t-1)$ learners via $\mathcal{L}^{(t-1)}$. Let Π_{t-1} be a distribution on such tasks. A level- t meta-experience is

$$\mathcal{G}^{(t)} = \{(U_{t-1}^{(j)}, \mathcal{D}^{(j)})\}_{j=1}^m, \quad U_{t-1}^{(j)} \stackrel{\text{i.i.d.}}{\sim} \Pi_{t-1},$$

where each $\mathcal{D}^{(j)}$ bundles the lower-level datasets needed by $U_{t-1}^{(j)}$. A level- t learner outputs a level- $(t-1)$ learner,

$$A^{(t)} : \bigcup_{m \geq 1} (\mathcal{T}_{t-1} \times \text{Data}_{t-1})^m \longrightarrow \{A^{(t-1)}\}, \quad \mathcal{G}^{(t)} \mapsto A_{\mathcal{G}^{(t)}}^{(t-1)},$$

and its level- t risk is defined recursively by

$$\mathcal{L}^{(t)}(A^{(t)}; \Pi_{t-1}) := \mathbb{E}_{\mathcal{G}^{(t)} \sim \Pi_{t-1}} \mathbb{E}_{U_{t-1} \sim \Pi_{t-1}} \mathbb{E}_{\mathcal{D} \sim U_{t-1}} \mathcal{L}^{(t-1)}(A^{(t)}(\mathcal{G}^{(t)}); U_{t-1}), \quad (3)$$

with base $\mathcal{L}^{(0)}$ (Definition 2.8) and $\mathcal{L}^{(1)}$ (Definition 2.10).

Example 2.13 (Iterated–Meta–Learning (level 2, everyday life)). A cooking school optimizes *how* instructors meta-learn across cohorts.

- (i) **Level-(1) tasks** U_1 . Each task specifies (a) a cohort’s dish distribution Π_0 (e.g., pastry-heavy vs. savory-heavy), and (b) an evaluation protocol (time-to-target, waste, taste scores) defining $\mathcal{L}^{(1)}$.
- (ii) **Meta-experience** $\mathcal{G}^{(2)}$. Over semesters, the school logs pairs $(U_1^{(j)}, \mathcal{D}^{(j)})$ comparing several meta-learning procedures (e.g., “pilot-then-gradient”, “case-base retrieval”, “video-first imitation”).
- (iii) **Output (a meta-learner selector)**. From $\mathcal{G}^{(2)}$, the school learns $A_{\mathcal{G}^{(2)}}^{(1)}$ that *selects/tunes* the best level-1 procedure $A^{(1)}$ given the cohort profile U_1 (e.g., choose “case-base retrieval” for pastry cohorts; otherwise choose “pilot-then-gradient” and tighten temperature priors).
- (iv) **Deployment**. For a *new* cohort U_1 , the level-2 learner $A^{(2)}$ outputs the tailored level-1 meta-learner $A_{\mathcal{G}^{(2)}}^{(1)}(U_1)$, which then rapidly adapts recipes for that cohort’s dishes, lowering the level-2 risk $\mathcal{L}^{(2)}$.

This is iterated meta-learning: $A^{(2)}$ learns to produce the *best meta-learner* for each evaluation context U_1 .

Definition 2.14 (Signature by depth for IML). Let $\Sigma_{\text{IML}}^{(0)}$ have carrier $H^{(0)} = \mathcal{H}$ and function/relations

$$\text{Pred}^{(0)} : \mathcal{H} \times \mathcal{X} \rightarrow \mathcal{Y}, \quad \text{EmpRisk}^{(0)} : (\mathcal{X} \times \mathcal{Y})^n \times \mathcal{H} \rightarrow \mathbb{R}_{\geq 0}, \quad \text{Train}^{(0)} : (\mathcal{X} \times \mathcal{Y})^* \rightarrow \mathcal{H}, \quad \text{Risk}^{(0)}(h, r).$$

For $t \geq 1$, set carrier $H^{(t)} = \text{Alg}^{(t-1)}$ (the set of level- $(t-1)$ learners) and add

$$\text{Train}^{(t)} : \text{Exp}^{(t)} \rightarrow \text{Alg}^{(t-1)}, \quad \text{Risk}^{(t)}(A^{(t-1)}, R) \text{ encoding } R = \mathcal{L}^{(t-1)}(A^{(t-1)}; \cdot).$$

Proposition 2.15 (Meta-operations on $\Sigma_{\text{IML}}^{(t)}$ -structures). For $\mathbf{C}_1^{(t)}, \dots, \mathbf{C}_k^{(t)}$ with carriers $H_i^{(t)}$:

- (i) **Tagged sum** $\Phi_{\oplus}^{(t)}$: set $H_{\oplus}^{(t)} := \prod_{i=1}^k H_i^{(t)}$ and define $\text{Train}_{\oplus}^{(t)}(\mathcal{G}^{(t)}, i) := \text{Train}_{\mathbf{C}_i}^{(t)}(\mathcal{G}^{(t)});$ interpret $\text{Risk}_{\oplus}^{(t)}$ componentwise on tagged inputs.

(ii) **Weighted ensemble (shared protocol)** $\Phi_{\text{ens}}^{(t)}$: if all inputs use the same evaluation protocol, set $H_{\text{ens}}^{(t)} := (\prod_{i=1}^k H_i^{(t)}) \times \Delta^{k-1}$ and

$$\text{Train}_{\text{ens}}^{(t)}(\mathcal{Z}^{(t)}) = \left(\text{Train}_{C_1}^{(t)}(\mathcal{Z}^{(t)}), \dots, \text{Train}_{C_k}^{(t)}(\mathcal{Z}^{(t)}), \hat{w}(\mathcal{Z}^{(t)}) \right),$$

where $\hat{w} \in \Delta^{k-1}$ minimizes an empirical proxy of $\mathcal{L}^{(t-1)}$.

Theorem 2.16 (IML generalizes Meta-Learning and forms an Iterated MetaStructure). For every $t \geq 1$:

(a) **(Generalization)** Fix a level- $(t-1)$ task U_{t-1}^* and let $\Pi_{t-1} := \delta_{U_{t-1}^*}$. If $A^{(t)}$ ignores $\mathcal{Z}^{(t)}$ and always returns a fixed level- $(t-1)$ learner $B^{(t-1)}$, then

$$\mathcal{L}^{(t)}(A^{(t)}; \Pi_{t-1}) = \mathbb{E}_{\mathcal{D}' \sim U_{t-1}^*} \mathcal{L}^{(t-1)}(B^{(t-1)}; U_{t-1}^*). \quad (4)$$

In particular, with $t = 1$ and $U_0^* = T_0$ (a single base task), IML reduces to Meta-Learning; fixing $B^{(0)}$ further reduces to the level-0 risk in Definition 2.8.

(b) **(Iterated MetaStructure)** Let $U^{(t)}$ be the class of $\Sigma_{\text{IML}}^{(t)}$ -structures and $\Phi^{(t)} \in \{\Phi_{\oplus}^{(t)}, \Phi_{\text{ens}}^{(t)}\}$ the meta-operations of Proposition 2.15. Then

$$\mathfrak{M}^{(t)} := (U^{(t)}, \{\Phi_{\oplus}^{(t)}, \Phi_{\text{ens}}^{(t)}\})$$

is a MetaStructure over $\Sigma_{\text{IML}}^{(t)}$ (uniform carrier constructors and symbol interpretations, isomorphism-invariant). Moreover, for $s < t$, the lift $\mathfrak{M}^{(s)\uparrow}$ (in the sense of Iterated MetaStructure) is a MetaStructure on $U^{(t)}$.

Proof. (a) *Generalization.* By the recursive definition (3) and $\Pi_{t-1} = \delta_{U_{t-1}^*}$,

$$\begin{aligned} \mathcal{L}^{(t)}(A^{(t)}; \Pi_{t-1}) &= \mathbb{E}_{\mathcal{Z}^{(t)} \sim \delta_{U_{t-1}^*}} \mathbb{E}_{U_{t-1} \sim \delta_{U_{t-1}^*}} \mathbb{E}_{\mathcal{D}' \sim U_{t-1}^*} \mathcal{L}^{(t-1)}(A^{(t)}(\mathcal{Z}^{(t)}); U_{t-1}^*) \\ &= \mathbb{E}_{\mathcal{D}' \sim U_{t-1}^*} \mathcal{L}^{(t-1)}(B^{(t-1)}; U_{t-1}^*), \end{aligned}$$

because $A^{(t)}(\mathcal{Z}^{(t)}) \equiv B^{(t-1)}$ by hypothesis. This proves (4).

Numerical sanity check (explicit). Take $t=1$, $\Pi_0 = \frac{1}{2}\delta_{T_1} + \frac{1}{2}\delta_{T_2}$, and fix a base learner $A^{(0)}$ with $\mathcal{L}^{(0)}(A^{(0)}; T_1) = 0.30$, $\mathcal{L}^{(0)}(A^{(0)}; T_2) = 0.50$. If $A^{(1)}$ always returns $A^{(0)}$, then

$$\mathcal{L}^{(1)}(A^{(1)}; \Pi_0) = \frac{1}{2} \cdot 0.30 + \frac{1}{2} \cdot 0.50 = 0.40.$$

With $\Pi_0 = \delta_{T_1}$ (Dirac), $\mathcal{L}^{(1)}(A^{(1)}; \delta_{T_1}) = 0.30$, which equals the level-0 risk on T_1 .

(b) *Iterated MetaStructure. Uniform carriers.* By Definition 2.14, each $\Sigma_{\text{IML}}^{(t)}$ -structure $C^{(t)}$ has carrier $H^{(t)} = \text{Alg}^{(t-1)}$. The meta-operations $\Phi_{\oplus}^{(t)}$ and $\Phi_{\text{ens}}^{(t)}$ construct $H_{\oplus}^{(t)} = \prod_i H_i^{(t)}$ and $H_{\text{ens}}^{(t)} = (\prod_i H_i^{(t)}) \times \Delta^{k-1}$, which depend only on input carriers (the constructor Γ_{ℓ} in the MetaStructure axioms).

Uniform symbol interpretations. The interpretations of $\text{Train}^{(t)}$ and $\text{Risk}^{(t)}$ under $\Phi_{\oplus}^{(t)}$ (tag routing) and $\Phi_{\text{ens}}^{(t)}$ (joint training with simplex weights \hat{w}) are given by the same recipes for any inputs, providing the required Λ_{ℓ}^f and Ξ_{ℓ}^R .

Isomorphism invariance (naturality). Let $\alpha_i : C_i^{(t)} \rightarrow D_i^{(t)}$ be $\Sigma_{\text{IML}}^{(t)}$ -isomorphisms: bijections $\alpha_i : H_i^{(t)} \rightarrow H_i'^{(t)}$ intertwining $\text{Train}^{(t)}$ and preserving $\text{Risk}^{(t)}$:

$$\text{Risk}_{C_i}^{(t)}(A^{(t-1)}, R) \iff \text{Risk}_{D_i}^{(t)}(\alpha_i(A^{(t-1)}), R).$$

Define $\alpha_{\oplus}^{(t)} := \prod_i \alpha_i$ and $\alpha_{\text{ens}}^{(t)} := (\prod_i \alpha_i) \times \text{id}_{\Delta^{k-1}}$. Then for all $\mathcal{Z}^{(t)}$,

$$\text{Train}_{D_{\oplus}}^{(t)} \circ \alpha_{\oplus}^{(t)} = \alpha_{\oplus}^{(t)} \circ \text{Train}_{C_{\oplus}}^{(t)}, \quad \text{Risk}^{(t)} \text{ is preserved likewise,}$$

hence $\Phi_{\oplus}^{(t)}(\alpha_1, \dots, \alpha_k)$ and $\Phi_{\text{ens}}^{(t)}(\alpha_1, \dots, \alpha_k)$ are induced isomorphisms. Therefore $\mathfrak{M}^{(t)}$ satisfies the MetaStructure axioms.

Lifted structures. For $s < t$, lifting acts symbolwise (replacing depth- s symbols by depth- t analogues and composing meta-operations), which preserves typing and naturality; hence $\mathfrak{M}^{(s)\uparrow}$ is a MetaStructure on $U^{(t)}$. \square

2.3. Iterated-Meta-analysis (Analysis of ... of Analysis)

Meta-analysis statistically combines results from multiple studies to estimate overall effects, assess heterogeneity, evaluate bias, and improve evidence, precision, reliability (cf. [44–47]). Iterated-Meta-analysis hierarchically aggregates meta-analyses across domains or time, modeling between-review dependencies, updating priors, synthesizing evidence streams, guiding decisions, policy, practice.

Definition 2.17 (Meta-analysis). Given K independent studies each providing an estimate $\hat{\theta}_i$ of a common effect θ with known/estimated variances $v_i > 0$, a (fixed-effect) *meta-analytic estimator* is the inverse-variance weighted mean

$$\hat{\theta}_{\text{FE}} = \frac{\sum_{i=1}^K w_i \hat{\theta}_i}{\sum_{i=1}^K w_i}, \quad w_i = \frac{1}{v_i}, \quad \text{Var}(\hat{\theta}_{\text{FE}}) = \frac{1}{\sum_i w_i}.$$

Allowing between-study heterogeneity $\tau^2 \geq 0$ (random effects) yields

$$\hat{\theta}_{\text{RE}} = \frac{\sum_{i=1}^K w_i^* \hat{\theta}_i}{\sum_{i=1}^K w_i^*}, \quad w_i^* = \frac{1}{v_i + \tau^2},$$

where τ^2 is estimated (e.g., method of moments or REML). Meta-analysis thus defines an estimator $\Psi(\{(\hat{\theta}_i, v_i)\}_{i=1}^K)$ that aggregates evidence to minimize mean squared error under the chosen model.

Example 2.18 (Meta-analysis (everyday life): estimating real fuel savings of a hybrid car). A shopper wants a single best estimate of the *MPG gain* of a hybrid vs. a comparable non-hybrid. They collect three independent road tests (treated as “studies”), each reporting an estimated MPG difference $\hat{\theta}_i$ and an uncertainty v_i (variance):

$$(\hat{\theta}_1, v_1) = (7.0, 4.0), \quad (\hat{\theta}_2, v_2) = (5.0, 1.0), \quad (\hat{\theta}_3, v_3) = (9.0, 9.0).$$

Using a fixed-effect meta-analysis (inverse-variance weights $w_i = 1/v_i$), the pooled estimate and variance are

$$\hat{\theta}_{\text{FE}} = \frac{\sum_i w_i \hat{\theta}_i}{\sum_i w_i} = \frac{0.25 \cdot 7 + 1 \cdot 5 + \frac{1}{9} \cdot 9}{0.25 + 1 + \frac{1}{9}} = \frac{7.75}{1.3611} \approx 5.694, \quad \text{Var}(\hat{\theta}_{\text{FE}}) = \frac{1}{\sum_i w_i} \approx 0.7347.$$

Thus $\text{SE} \approx 0.857$ and a 95% CI is $5.694 \pm 1.96 \cdot 0.857 \approx [4.02, 7.37]$ MPG. The shopper concludes the hybrid yields about 6 MPG more (precision improved by pooling the three tests).

Definition 2.19 (Iterated Meta-analysis (depth t)). At level 0, a *study* provides an estimate-variance pair $(\hat{\theta}_i^{(0)}, v_i^{(0)})$. At level 1 (ordinary meta-analysis, cf. Definition 2.17), given pairs $\{(\hat{\theta}_i^{(0)}, v_i^{(0)})\}_{i=1}^K$, the fixed-effect (FE) pooled pair is

$$\hat{\theta}^{(1)} = \frac{\sum_{i=1}^K w_i^{(1)} \hat{\theta}_i^{(0)}}{\sum_{i=1}^K w_i^{(1)}}, \quad (v^{(1)})^{-1} = \sum_{i=1}^K w_i^{(1)}, \quad w_i^{(1)} := \frac{1}{v_i^{(0)}}.$$

(With random effects, replace $w_i^{(1)}$ by $1/(v_i^{(0)} + \tau_1^2)$ with a chosen $\tau_1^2 \geq 0$.)

For $t \geq 2$, suppose we are given a finite family of *children nodes* indexed by j , each child providing a (level $t-1$) meta-analytic pair $(\hat{\theta}_j^{(t-1)}, v_j^{(t-1)})$. The level- t pooled pair is defined recursively by

$$\hat{\theta}^{(t)} := \frac{\sum_j w_j^{(t)} \hat{\theta}_j^{(t-1)}}{\sum_j w_j^{(t)}}, \quad (v^{(t)})^{-1} := \sum_j w_j^{(t)}, \quad w_j^{(t)} := \frac{1}{v_j^{(t-1)} + \tau_t^2}, \quad (5)$$

with *depth- t heterogeneity* $\tau_t^2 \geq 0$ (set $\tau_t^2=0$ for FE). We call any estimator produced by the recursion (5) an *Iterated Meta-analytic estimator* of depth t .

Remark 2.20 (Two-stage FE equals one-stage FE). If every child level uses FE ($\tau_s^2 = 0$ for all s), then the two-stage FE obtained by first pooling within disjoint groups and then pooling across groups equals the single-stage FE pooling of all studies at once; see the algebra in the proof of Theorem 2.24.

Example 2.21 (Iterated Meta-analysis (everyday life): pooling regional meta-analyses). A national consumer group wants a single summary across *regional* syntheses. Each region has already meta-analyzed multiple local road tests (level 1), yielding a pooled effect and variance:

$$\text{Region A: } (\hat{\theta}_A^{(1)}, v_A^{(1)}) = (6.0, 0.50), \quad \text{Region B: } (\hat{\theta}_B^{(1)}, v_B^{(1)}) = (4.8, 0.80).$$

At level 2, they pool these *meta-analytic* pairs using fixed-effect weights $W_A = 1/0.50 = 2$ and $W_B = 1/0.80 = 1.25$:

$$\hat{\theta}^{(2)} = \frac{2 \cdot 6.0 + 1.25 \cdot 4.8}{2 + 1.25} = \frac{12.0 + 6.0}{3.25} \approx 5.538, \quad \text{Var}(\hat{\theta}^{(2)}) = \frac{1}{2 + 1.25} \approx 0.3077.$$

Hence $\text{SE} \approx 0.555$ and a 95% CI is $5.538 \pm 1.96 \cdot 0.555 \approx [4.45, 6.63]$ MPG. This *iterated* pooling coherently combines already-pooled regional evidence into one national estimate, mirroring Definition 2.19 with $t = 2$.

Definition 2.22 (Signature by depth for meta-analysis). Let $\Sigma_{\text{MA}}^{(0)}$ have carrier $H^{(0)} = \{(\hat{\theta}, v) \in \mathbb{R} \times (0, \infty)\}$ and symbols

$$\text{Est}^{(0)} : \text{StudyData} \rightarrow H^{(0)}, \quad \text{Var}^{(0)} : H^{(0)} \rightarrow (0, \infty).$$

For $t \geq 1$, set the carrier $H^{(t)} = H^{(0)}$ (each node is summarized by an estimate-variance pair) and add a *combiner*

$$\text{Combine}^{(t)} : (H^{(t-1)})^k \times \mathbb{R}_{\geq 0} \rightarrow H^{(t)}, \quad ((\hat{\theta}_j^{(t-1)}, v_j^{(t-1)})_{j=1}^k, \tau_t^2) \mapsto (\hat{\theta}^{(t)}, v^{(t)})$$

interpreted by (5).

Proposition 2.23 (Meta-operations and naturality). Let $\mathbf{C}_1^{(t)}, \dots, \mathbf{C}_k^{(t)}$ be $\Sigma_{\text{MA}}^{(t)}$ -structures (nodes) with carriers $H^{(t)} = H^{(0)}$. Define the following meta-operations on $U^{(t)} \subseteq \text{Str}_{\Sigma_{\text{MA}}^{(t)}}$:

- (i) **Tagged sum** $\Phi_{\oplus}^{(t)}$: the carrier constructor is the Cartesian product of carriers; $\text{Combine}^{(t)}$ is applied to the selected tag; variances and estimates pass through componentwise.
- (ii) **Weighted ensemble** $\Phi_{\text{ens}}^{(t)}$: when all inputs share the same model for τ_t^2 , set the carrier constructor to the product-with-simplex $H_{\text{ens}}^{(t)} = (H^{(t)})^k \times \Delta^{k-1}$ and interpret

$$\text{Combine}_{\text{ens}}^{(t)}((\hat{\theta}_j^{(t-1)}, v_j^{(t-1)})_{j=1}^k, \tau_t^2) = \left(\frac{\sum_j w_j^{(t)} \hat{\theta}_j^{(t-1)}}{\sum_j w_j^{(t)}}, \left(\sum_j w_j^{(t)} \right)^{-1} \right), \quad w_j^{(t)} = \frac{1}{v_j^{(t-1)} + \tau_t^2}.$$

Both operations are isomorphism-invariant under any permutation σ of the input indices (relabeling studies), since the formulas for $(\hat{\theta}^{(t)}, v^{(t)})$ are symmetric in the multiset of pairs $\{(\hat{\theta}_j^{(t-1)}, v_j^{(t-1)})\}$.

Theorem 2.24 (IMA generalizes MA and forms an Iterated MetaStructure). For every $t \geq 1$:

- (a) (**Generalization**) If $t = 1$ and $\tau_1^2 = 0$, Definition 2.19 reduces to the fixed-effect meta-analytic estimator (Definition 2.17). More generally, with $\tau_s^2 \equiv 0$ at all levels and grouping of base studies into any disjoint family, the two-stage FE estimator equals the one-stage FE estimator.
- (b) (**Iterated MetaStructure**) With $U^{(t)}$ the class of $\Sigma_{\text{MA}}^{(t)}$ -structures and meta-operations $\{\Phi_{\oplus}^{(t)}, \Phi_{\text{ens}}^{(t)}\}$ from Proposition 2.23,

$$\mathfrak{M}^{(t)} := (U^{(t)}, \{\Phi_{\oplus}^{(t)}, \Phi_{\text{ens}}^{(t)}\})$$

is a MetaStructure over $\Sigma_{\text{MA}}^{(t)}$ (uniform carrier constructors and symbol interpretations, natural under isomorphisms). If $s < t$, the lift $\mathfrak{M}^{(s)\uparrow}$ (in the sense of Iterated MetaStructure) is again a MetaStructure on $U^{(t)}$.

Proof. (a) *Generalization and explicit algebra.* The $t=1$ case with $\tau_1^2=0$ is exactly the FE formula in Definition 2.17. For the two-stage FE equivalence, partition the K base studies into disjoint groups $g = 1, \dots, G$. Within each group g , let $w_{gi} := 1/v_{gi}$ and $W_g := \sum_{i \in g} w_{gi}$. The group FE estimate and variance are

$$\hat{\theta}_g = \frac{\sum_{i \in g} w_{gi} \hat{\theta}_{gi}}{W_g}, \quad \text{Var}(\hat{\theta}_g) = \frac{1}{W_g}.$$

At the second stage, use weights W_g (the inverse variances) to form

$$\hat{\theta}_{\text{two-stage}} = \frac{\sum_{g=1}^G W_g \hat{\theta}_g}{\sum_{g=1}^G W_g} = \frac{\sum_g W_g \cdot (\sum_{i \in g} w_{gi} \hat{\theta}_{gi} / W_g)}{\sum_g W_g} = \frac{\sum_g \sum_{i \in g} w_{gi} \hat{\theta}_{gi}}{\sum_g \sum_{i \in g} w_{gi}} = \hat{\theta}_{\text{one-stage}}.$$

Likewise, the variance satisfies $\text{Var}(\hat{\theta}_{\text{two-stage}}) = 1/\sum_g W_g = 1/\sum_g \sum_{i \in g} w_{gi} = \text{Var}(\hat{\theta}_{\text{one-stage}})$.

Numerical check. Consider three studies split into two groups:

$$(\hat{\theta}_{11}, v_{11}) = (1.00, 0.04), \quad (\hat{\theta}_{12}, v_{12}) = (1.20, 0.09), \quad (\hat{\theta}_{21}, v_{21}) = (0.80, 0.01).$$

Within group $g=1$: $w_{11} = 1/0.04 = 25$, $w_{12} = 1/0.09 \approx 11.111$, so $W_1 = 36.111$ and

$$\hat{\theta}_1 = \frac{25 \cdot 1.00 + 11.111 \cdot 1.20}{36.111} \approx \frac{25 + 13.333}{36.111} = 1.06154, \quad \text{Var}(\hat{\theta}_1) = 1/W_1 \approx 0.02769.$$

Group $g=2$: $W_2 = 1/0.01 = 100$, $\hat{\theta}_2 = 0.80$. Two-stage FE:

$$\hat{\theta}_{\text{two-stage}} = \frac{36.111 \cdot 1.06154 + 100 \cdot 0.80}{36.111 + 100} = \frac{38.333 + 80}{136.111} = 0.869388, \quad \text{Var} = 1/136.111 = 0.00734694.$$

One-stage FE over all three studies:

$$\hat{\theta}_{\text{one-stage}} = \frac{25 \cdot 1.00 + 11.111 \cdot 1.20 + 100 \cdot 0.80}{25 + 11.111 + 100} = \frac{25 + 13.333 + 80}{136.111} = 0.869388, \quad \text{Var} = 0.00734694,$$

which matches exactly.

(b) *Iterated MetaStructure. Uniform carrier constructors.* In Definition 2.22, all carriers are $H^{(0)}$ (estimate-variance pairs). For any finite input family, the tagged product and the product-with-simplex (when present) depend only on carriers, supplying the Γ_ℓ required by MetaStructure.

Uniform symbol interpretations. The interpretation of $\text{Combine}^{(t)}$ is given by the single recipe (5), independent of representatives; this provides the uniform Λ_ℓ^f (functions) and Ξ_ℓ^R (relations) in the MetaStructure axioms.

Isomorphism invariance (naturality). Any permutation σ of inputs induces a relabeling isomorphism (bijection on carriers) under which the multiset $\{(\hat{\theta}_j^{(t-1)}, v_j^{(t-1)})\}$ is unchanged; since (5) is symmetric in this multiset, the output pair $(\hat{\theta}^{(t)}, v^{(t)})$ is invariant, yielding induced isomorphisms $\Phi_{\oplus}^{(t)}(\sigma)$ and $\Phi_{\text{ens}}^{(t)}(\sigma)$. Therefore $\mathfrak{M}^{(t)}$ is a MetaStructure.

Lifted structures. If $s < t$, the lift $\mathfrak{M}^{(s)\uparrow}$ replaces depth- s symbols by their depth- t counterparts and composes meta-operations componentwise; typing and naturality are preserved, hence $\mathfrak{M}^{(s)\uparrow}$ is again a MetaStructure on $U^{(t)}$. \square

2.4. Iterated Metadiscourse (Discussion about ... about a discussion)

Metadiscourse refers to language used to organize, evaluate, and guide interpretation of discourse, marking structure, stance, hedging, emphasis, or reader engagement beyond subject content [48–52]. Iterated Metadiscourse recursively applies discourse about discourse to itself, layering commentary on rhetorical devices, coherence, and stance, formalized through Iterated-MetaStructure representation.

Definition 2.25 (Metadiscourse). Fix a two-sorted semantics with an *object world* W and an *utterance world* U . Let \mathcal{L} be a language whose sentences have denotations in either Prop_W (propositions about W) or Prop_U (propositions about U). For a finite discourse $D \subseteq \text{Sent}(\mathcal{L})$, write $\llbracket \cdot \rrbracket_W : \text{Sent}(\mathcal{L}) \rightarrow \text{Prop}_W$ and $\llbracket \cdot \rrbracket_U : \text{Sent}(\mathcal{L}) \rightarrow \text{Prop}_U$. A sentence φ is *metadiscursive* (w.r.t. D) iff its truth depends only on rhetorical/structural or stance features of D , i.e.

$$\varphi \in \text{MD}(D) \iff \exists F : \text{Struct}(D) \rightarrow \{\top, \perp\} \text{ such that } \llbracket \varphi \rrbracket_U = F(\text{Struct}(D)),$$

where $\text{Struct}(D)$ encodes text-internal features (ordering, sectioning, connectives, hedges/boosters, attitude markers, cue phrases).

Example 2.26 (Metadiscourse (everyday email)). A manager sends a short email to staff:

“First, the office will open at 10am tomorrow. Second, client calls move to Friday. Finally, please confirm by 5pm. Frankly, this is not ideal, but it should work.”

Here the words *First/Second/Finally* are **frame/sequence markers** that organize the message; *please confirm* is an **engagement directive**; *Frankly* is an **attitude marker**. Consider the metadiscursive claim:

φ : “The email explicitly signals its three-part structure and expresses stance.”

The truth of φ depends only on the rhetorical features $\text{Struct}(D)$ (presence of sequence markers and stance markers), not on whether the office actually opens at 10am in the world W . Hence $\varphi \in \text{MD}(D)$ as in the definition.

Definition 2.27 (Iterated Metadiscourse of depth t). Let $D^{(0)} \subseteq \text{Sent}(\mathcal{L})$ be a base (object-level) discourse. For $k \geq 1$ define the k -th *metadiscourse layer*

$$D^{(k)} := \text{MD}(D^{(k-1)}) = \{ \varphi \in \text{Sent}(\mathcal{L}) : \exists F_k : \text{Struct}(D^{(k-1)}) \rightarrow \{\top, \perp\}, \llbracket \varphi \rrbracket_U^{(k)} = F_k(\text{Struct}(D^{(k-1)})) \},$$

where $\llbracket \cdot \rrbracket_U^{(k)} : \text{Sent}(\mathcal{L}) \rightarrow \text{Prop}_U^{(k)}$ denotes level- k utterance-world denotations (propositions about the $(k-1)$ -level discourse). The *iterated metadiscourse tower of depth t* is

$$\mathbb{D}^{(\leq t)} := (D^{(0)}, D^{(1)}, \dots, D^{(t)}), \quad \text{and its closure } \bar{D}^{(\leq t)} := \bigcup_{k=0}^t D^{(k)}.$$

Remark 2.28 (Typing discipline). $D^{(0)}$ is typed by Prop_W via $\llbracket \cdot \rrbracket_W$, while $D^{(k)}$ (for $k \geq 1$) is typed by $\text{Prop}_U^{(k)}$ via $\llbracket \cdot \rrbracket_U^{(k)}$; that is, $D^{(k)}$ makes claims about the structure/stance of $D^{(k-1)}$, not about W directly.

Example 2.29 (Iterated Metadiscourse (feedback about feedback)). **Level 0 (discourse $D^{(0)}$)**. A student writes: “In this report, I first outline the method; however, the results may suggest a limitation.” This contains a **frame marker** (*first*), a **contrastive connective** (*however*), and a **hedge** (*may*).

Level 1 (metadiscourse $D^{(1)} = \text{MD}(D^{(0)})$). A TA comments: “Good structure signal with ‘first’; your stance is cautious due to ‘may’. Consider adding a ‘finally’ to close the sequence.” These remarks are **about** the student’s rhetorical devices, hence metadiscourse on $D^{(0)}$.

Level 2 (iterated metadiscourse $D^{(2)} = \text{MD}(D^{(1)})$). A course coordinator replies to the TA: “Your feedback prioritizes hedging detection over coherence guidance; add explicit transition advice (e.g., ‘next’, ‘in sum’).” This sentence evaluates the structure and stance-focus of the TA’s feedback, so its truth depends on $\text{Struct}(D^{(1)})$ (which features the TA addressed and which it omitted), not on the student’s scientific content. Formally, if

ψ : “The TA’s feedback emphasizes hedges more than transitions.”,

then $\psi \in \text{MD}(D^{(1)})$, witnessing a depth-2 (iterated) metadiscourse claim.

Theorem 2.30 (Iterated Metadiscourse generalizes Metadiscourse). *For any base discourse $D^{(0)}$, the depth-1 tower $(D^{(0)}, D^{(1)})$ satisfies $D^{(1)} = \text{MD}(D^{(0)})$, hence Definition 2.27 reduces to the usual notion of metadiscourse at $t = 1$.*

Proof. By Definition 2.27 with $k = 1$ we set $D^{(1)} = \{\varphi : \llbracket \varphi \rrbracket_U^{(1)} = F_1(\text{Struct}(D^{(0)}))\}$, which matches the “Metadiscourse” definition with $D := D^{(0)}$ (only the superscript on $\llbracket \cdot \rrbracket_U$ distinguishes the level). Thus $D^{(1)} = \text{MD}(D^{(0)})$. \square

Definition 2.31 (Metadiscourse as an Iterated MetaStructure). Let Σ be a single-sorted signature whose structures encode finite discourses equipped with their internal annotations:

$$\mathbf{D} = (H, \text{ord}, \text{sec}, \text{conn}, \text{hedge}, \text{att}),$$

where $H \subseteq \text{Sent}(\mathcal{L})$ and the symbols capture, respectively, linear order, sectioning, discourse connectives, hedges/boosters, and attitude markers. Let $U^{(0)}$ be the class of such Σ -structures (object-level discourses), and for $k \geq 0$ define the meta-operation

$$\Phi : U^{(k)} \longrightarrow U^{(k+1)}, \quad \Phi(\mathbf{D}) := \mathbf{MD}(\mathbf{D}),$$

where $\mathbf{MD}(\mathbf{D})$ is the Σ -structure whose carrier is $D^{(k+1)} = \text{MD}(H)$ and whose symbols are uniformly computed from $\text{Struct}(H)$ (e.g., ord on the metalevel inherits the canonical order induced by the generating features referenced in F_k).

Proposition 2.32 (Naturality / isomorphism invariance). *If $\alpha : \mathbf{D} \xrightarrow{\cong} \mathbf{D}'$ is a Σ -isomorphism preserving Struct, then there exists an induced isomorphism $\Phi(\alpha) : \Phi(\mathbf{D}) \xrightarrow{\cong} \Phi(\mathbf{D}')$. Hence Φ is isomorphism-invariant (natural).*

Proof. Because each $\varphi \in \text{MD}(H)$ is defined solely by a Boolean functional $F_k(\text{Struct}(H))$, and α preserves Struct by assumption, the truth of every metadiscursive sentence is preserved under α . Mapping each metadiscursive sentence in $\Phi(\mathbf{D})$ to its α -transported counterpart in $\Phi(\mathbf{D}')$ yields a bijection commuting with all Σ -symbols by construction. \square

Theorem 2.33 (Representation as an Iterated MetaStructure). *Let $t \in \mathbb{N}$. The tower*

$$\mathfrak{M}^{(t)} := (U^{(0)}, U^{(1)}, \dots, U^{(t)}; \Phi), \quad \text{with } U^{(k+1)} := \Phi(U^{(k)}),$$

is an Iterated MetaStructure of depth t : repeated application of the natural meta-operation Φ produces the hierarchy of discourses-about-discourses, and the lifts Φ^\dagger coincide with Φ by definition. Moreover, $t = 1$ recovers ordinary Metadiscourse (Theorem 2.30).

Proof. By the Proposition, Φ is an isomorphism-invariant constructor on $U^{(k)}$, so it is a valid meta-operation in the sense of MetaStructure. Define $U^{(k+1)} := \Phi(U^{(k)})$ and iterate t times; this realizes the lifting procedure that builds an Iterated MetaStructure. Unwinding the recursion shows that objects in $U^{(k)}$ encode level- k metadiscourse over the base, and the case $t = 1$ gives exactly $\text{MD}(D^{(0)})$. \square

2.5. Iterated Metaphilosophy (Philosophy of ... of Philosophy)

Philosophy systematically investigates fundamental questions about reality, knowledge, value, mind, language, logic, meaning, methodology, and reasoning. evidence, argument, clarity, wisdom (cf.[53–56]). Metaphilosophy critically examines philosophy itself: its aims, methods, questions, boundaries, progress, evaluation standards, practices, institutions, pedagogy, communication. history, value, limitations(cf.[57–60]). Iterated metaphilosophy recursively studies higher-order layers: metaphilosophies evaluating other metaphilosophies, formalizing reflexive methods, coherence, convergence, governance, incentive structures. dynamics, impact.

Definition 2.34 (Metaphilosophy). Let a (first-order) *philosophical theory* be a quadruple

$$\Phi = (Q, \text{Ans}, \text{Meth}, \text{Norm})$$

, where Q is a set of questions, Ans a set of candidate answers, Meth a set of admissible methods, and Norm a set of evaluation norms (e.g. validity, clarity, fruitfulness). A *metaphilosophical theory* Φ^\uparrow is a theory whose non-logical symbols range over the components of Φ and whose axioms/theorems are propositions about them, e.g.

$$\text{WellFormed}(q), \quad \text{Admissible}(m), \quad \text{Adequate}(n, q, a),$$

together with rules that compare or constrain $(Q, \text{Meth}, \text{Norm})$. Formally, if \mathcal{L}_Φ is the language of Φ , a metaphilosophy is any theory in a meta-language $\mathcal{L}_\Phi^\uparrow$ that *quantifies over and predicates of* $Q, \text{Ans}, \text{Meth}, \text{Norm}$.

Example 2.35 (Metaphilosophy (departmental policy in practice)). A university philosophy department meets to design its research and teaching policy.

- **Object level** $\Phi^{(0)}$: faculty pursue questions such as “Do moral facts supervene on natural facts?” using methods like thought experiments, formal modeling, or historical analysis, and they publish candidate answers.
- **Meta level** $\Phi^{(1)} = \uparrow(\Phi^{(0)})$: the committee explicitly debates *about* the components of $\Phi^{(0)}$:
 1. Q : Which questions count as central (e.g., priority to public-impact ethics vs. abstract metaphysics)?
 2. Meth : Which methods are *admissible* (e.g., allow experimental philosophy surveys; require formal clarity for modality debates)?
 3. Norm : Which evaluation norms govern papers and courses (e.g., transparency of argument maps, reproducible data for X-phi, fruitfulness for cross-field collaboration)?

The committee’s resolutions (e.g., “*Admissible(survey)*”; “*WellFormed(question drafts require argument maps)*”) are propositions about $Q, \text{Meth}, \text{Norm}$ of $\Phi^{(0)}$, hence constitute a concrete metaphilosophical theory in everyday departmental governance.

Definition 2.36 (Iterated Metaphilosophy of depth t). Fix a base philosophical theory

$$\Phi^{(0)} = (Q^{(0)}, \text{Ans}^{(0)}, \text{Meth}^{(0)}, \text{Norm}^{(0)})$$

. Define a meta-operator

$$\uparrow: \Phi^{(k)} \mapsto \Phi^{(k+1)} := (Q^{(k+1)}, \text{Ans}^{(k+1)}, \text{Meth}^{(k+1)}, \text{Norm}^{(k+1)}),$$

where each component of level $k+1$ is *about* level k :

$$\begin{aligned} Q^{(k+1)} &:= \{ \text{questions about } (Q^{(k)}, \text{Ans}^{(k)}, \text{Meth}^{(k)}, \text{Norm}^{(k)}) \}, \\ \text{Ans}^{(k+1)} &:= \{ \text{propositions/claims answering elements of } Q^{(k+1)} \}, \\ \text{Meth}^{(k+1)} &:= \{ \text{meta-methods that evaluate/compare } \text{Meth}^{(k)} \text{ and } \text{Norm}^{(k)} \}, \\ \text{Norm}^{(k+1)} &:= \{ \text{norms to appraise elements of } \text{Ans}^{(k+1)} \text{ and } \text{Meth}^{(k+1)} \}. \end{aligned}$$

For $t \in \mathbb{N}$, the *iterated metaphilosophy tower of depth t* is

$$\blacksquare^{(\leq t)} := (\Phi^{(0)}, \Phi^{(1)}, \dots, \Phi^{(t)}), \quad \text{with } \Phi^{(k+1)} = \uparrow\uparrow (\Phi^{(k)}).$$

When needed, we write $\mathcal{L}^{(k)}$ for the (meta)-language in which $\Phi^{(k)}$ is formulated, so that $\mathcal{L}^{(k+1)}$ extends $\mathcal{L}^{(k)}$ by symbols that range over the components of $\Phi^{(k)}$.

Remark 2.37 (Level discipline). $\Phi^{(0)}$ makes claims *about the world*; $\Phi^{(1)}$ makes claims *about*

$$(Q^{(0)}, \text{Ans}^{(0)}, \text{Meth}^{(0)}, \text{Norm}^{(0)})$$

; in general $\Phi^{(k)}$ makes claims about $\Phi^{(k-1)}$. Cross-level conservativity or reflection principles may be added, but are not required by the definition.

Example 2.38 (Iterated Metaphilosophy (governing the governors)). A national philosophy association evaluates *departments' meta-policies* to issue accreditation.

- **Level 1** $\Phi^{(1)}$: each department has a metaphilosophical framework (e.g., “pluralist admissibility of methods with clarity-and-impact norms”) regulating its object-level research $\Phi^{(0)}$.
- **Level 2** $\Phi^{(2)} = \uparrow\uparrow (\Phi^{(1)})$: the association forms an oversight panel that compares and audits those *metaphilosophies*. Its questions and rules are *about* $Q^{(1)}, \text{Meth}^{(1)}, \text{Norm}^{(1)}$:
 1. $Q^{(2)}$: “Do the departments’ admissibility rules systematically bias against certain subfields or methods?”
 2. $\text{Meth}^{(2)}$: “Use rubric-based audits, citation-network analysis, and stakeholder interviews to evaluate $\text{Norm}^{(1)}$.”
 3. $\text{Norm}^{(2)}$: “Prefer metaphilosophies that ensure transparency of review, methodological pluralism, and measurable educational outcomes.”

The accreditation decision (e.g., “Adequate($\text{Norm}^{(2)}$, bias-robustness, Dept A’s $\Phi^{(1)}$)”) is a claim about a *metaphilosophy* rather than about object-level philosophy. Thus everyday accreditation practice exemplifies iterated metaphilosophy: philosophy *about* philosophy *about* philosophy.

Theorem 2.39 (Iterated Metaphilosophy generalizes Metaphilosophy). *For any base $\Phi^{(0)}$, the depth-1 tower $(\Phi^{(0)}, \Phi^{(1)})$ satisfies $\Phi^{(1)} = \uparrow\uparrow (\Phi^{(0)})$ and coincides with the usual notion of metaphilosophy.*

Proof. By Definition 2.36, $\Phi^{(1)}$ has a language $\mathcal{L}^{(1)}$ whose non-logical symbols range over the components of $\Phi^{(0)}$, and its sentences are propositions *about* those components. This is exactly the “Metaphilosophy (recalled)” notion. \square

Definition 2.40 (Encoding as an Iterated MetaStructure). Let Σ_{phil} be a single-sorted signature with unary predicates Q, A, M, N picking out questions, answers, methods, and norms inside a carrier H , and relation symbols

$$\text{Admissible}(m), \text{WellFormed}(q), \text{Adequate}(n, q, a), \text{Prefer}(n, a, a'),$$

together with any additional (fixed) vocabulary needed to represent a philosophical theory internally. A *level-0 object* is a Σ_{phil} -structure

$$\blacksquare = (H; Q^{\blacksquare}, A^{\blacksquare}, M^{\blacksquare}, N^{\blacksquare}; \text{Admissible}^{\blacksquare}, \text{WellFormed}^{\blacksquare}, \text{Adequate}^{\blacksquare}, \text{Prefer}^{\blacksquare}),$$

interpreting the components of a philosophical theory on H . Define a *meta-operation*

$$\Phi_* : U^{(k)} \longrightarrow U^{(k+1)}, \quad \blacksquare \longmapsto \blacksquare^\uparrow,$$

where the carrier of \blacksquare^\uparrow consists of well-formed formulas in the meta-language that *refer to* the relations of \blacksquare (e.g. encodings of Admissible(m), Adequate(n, q, a), etc.), and where each predicate/relation of Σ_{phil} on the meta-level is computed by a fixed, isomorphism-invariant recipe from the corresponding data of \blacksquare . Set $U^{(0)} := \{\text{all } \Sigma_{\text{phil}}\text{-structures}\}$ and $U^{(k+1)} := \Phi_*(U^{(k)})$.

Proposition 2.41 (Isomorphism invariance (naturality)). *If $\alpha : \blacksquare \xrightarrow{\cong} \blacksquare$ is a Σ_{phil} -isomorphism, then there exists an induced isomorphism $\Phi_*(\alpha) : \blacksquare^\uparrow \xrightarrow{\cong} \blacksquare^\uparrow$.*

Proof. By construction, the carrier of \blacksquare^\uparrow is the set of meta-sentences obtained from the *invariantly* defined relations of \blacksquare ; transporting along α preserves truth of such sentences because α preserves all Σ_{phil} -relations. Thus mapping each meta-sentence to its α -transported counterpart yields a bijection that commutes with the interpretations of the meta-level predicates and relations. \square

Theorem 2.42 (Representation as an Iterated MetaStructure). *For every depth $t \in \mathbb{N}$, the tower*

$$\mathfrak{M}^{(t)} := (U^{(0)}, U^{(1)}, \dots, U^{(t)}; \Phi_*), \quad U^{(k+1)} := \Phi_*(U^{(k)}),$$

is an Iterated MetaStructure of depth t whose objects encode $\Phi^{(0)}, \Phi^{(1)}, \dots, \Phi^{(t)}$. Moreover, the case $t = 1$ recovers ordinary metaphilosophy (Theorem 2.39).

Proof. By Proposition 2.41, Φ_* is an isomorphism-invariant constructor on $U^{(k)}$, hence a valid meta-operation in the sense of MetaStructure. Iterating Φ_* produces the desired hierarchy; unpacking the definitions shows that $U^{(k)}$ consists exactly of encodings of level- k metaphilosophical theories about level- $(k-1)$. For $t = 1$, $U^{(1)} = \Phi_*(U^{(0)})$ coincides with the usual metaphilosophy over $U^{(0)}$, establishing the generalization. \square

2.6. Iterated Metaknowledge (Knowledge of ... of Knowledge)

Metaknowledge concerns knowledge about knowledge: its sources, structures, justification, uncertainty, sharing, retrieval, governance, evolution, applications, and limitations across agents communities (cf.[61–66]). Iterated metaknowledge recursively studies layers of knowledge about knowledge, modeling higher-order beliefs, reflexive reasoning, communication protocols, incentives, and governance dynamics.

Definition 2.43 (Metaknowledge). Let $\mathcal{M} = (S, \{R_i\}_{i \in I}, V^{(0)})$ be a Kripke model for agents $i \in I$ over a base propositional language $\mathcal{L}^{(0)}$ with valuation $V^{(0)} : \text{Prop} \rightarrow \mathcal{P}(S)$. For φ in the Boolean closure of $\mathcal{L}^{(0)}$ under the unary modalities K_i , define the *order*

$$\text{ord}(\varphi) = 0 \text{ if } \varphi \text{ has no } K_i; \quad \text{ord}(K_i\psi) = \text{ord}(\psi) + 1; \quad \text{ord}(\psi \circ \chi) = \max\{\text{ord}(\psi), \text{ord}(\chi)\}.$$

A formula is *meta-knowledge of order* $n \geq 1$ iff $\text{ord}(\varphi) = n$.

Example 2.44 (Metaknowledge (everyday transactive memory)). In a household, a teenager knows that their mother knows the trusted plumber's contact. Let p denote the base proposition "the recommended plumber's phone number is #". The teen's state can be expressed as

$$K_{\text{teen}} K_{\text{mom}} p,$$

i.e., knowledge *about* someone else's knowledge (metaknowledge). Practically, when a leak occurs, the teen does not search the web but immediately calls the mother, leveraging knowledge-of-where-the-knowledge-is stored (a family 'who-knows-what' map).

Definition 2.45 (Language/semantics tower). Define a family of languages $(\mathcal{L}^{(t)})_{t \geq 0}$ by

$$\mathcal{L}^{(0)} \text{ given, } \quad \mathcal{L}^{(t+1)} := \text{Bool}(\{K_i \psi \mid i \in I, \psi \in \mathcal{L}^{(t)}\}).$$

For each t , define satisfaction $\models^{(t)} \subseteq S \times \mathcal{L}^{(t)}$ inductively: Boolean clauses as usual, and

$$\mathcal{M}, s \models^{(t+1)} K_i \psi \iff \forall s' (s R_i s' \Rightarrow \mathcal{M}, s' \models^{(t)} \psi), \quad \psi \in \mathcal{L}^{(t)}.$$

Definition 2.46 (Iterated Metaknowledge of depth t). Fix $t \in \mathbb{N}$. The *iterated metaknowledge structure of depth t* on \mathcal{M} is

$$\mathbf{MK}^{(\leq t)}(\mathcal{M}) := (\mathcal{M}; \models^{(0)}, \models^{(1)}, \dots, \models^{(t)}),$$

equipped with the language tower $(\mathcal{L}^{(0)}, \dots, \mathcal{L}^{(t)})$ from Definition 2.45. The set of all depth- t meta-knowledge formulae is $\mathcal{L}^{(\leq t)} := \bigcup_{k=1}^t \mathcal{L}^{(k)}$.

Example 2.47 (Iterated Metaknowledge (layered expertise in a project team)). In a startup, the on-call engineer A knows that the tech lead B knows that the compliance officer C knows whether a specific clause q permits storing user logs for 90 days. Formally,

$$K_A K_B K_C q.$$

This *iterated* (third-order) metaknowledge determines the escalation path: A pages B (who will consult C) instead of making an ad hoc decision. If, moreover, B knows that A knows that B knows that C knows q (and this becomes common knowledge within the on-call playbook), hand-offs become faster and errors fewer, illustrating how higher-order knowledge about knowledge improves coordination under time pressure.

Proposition 2.48 (Generalization). *At $t = 1$, $\mathcal{L}^{(\leq 1)} = \mathcal{L}^{(1)}$ is exactly the set of order-1 metaknowledge statements (“knowledge about base facts”), and all order- n metaknowledge with $n \leq t$ belongs to $\mathcal{L}^{(\leq t)}$. Hence Iterated Metaknowledge of depth t generalizes (single-step) Metaknowledge.*

Proof. By construction, $\mathcal{L}^{(1)}$ is the Boolean closure of $\{K_i \psi : \psi \in \mathcal{L}^{(0)}\}$, which are exactly order-1 forms. The recursion for $\mathcal{L}^{(t)}$ increases outermost modal depth by one, so every order- n formula lies in some $\mathcal{L}^{(n)} \subseteq \mathcal{L}^{(\leq t)}$ whenever $n \leq t$. \square

Let Σ_{ep} be the single-sorted signature with carrier S and relation symbols $\{R_i\}_{i \in I}$ together with, for each $t \geq 0$, a binary predicate $\text{Sat}^{(t)}(s, \varphi)$ intended as “ $s \models^{(t)} \varphi$ ”. Consider the class

$$U^{(0)} := \{ \mathbf{E} = (S, \{R_i\}_{i \in I}, V^{(0)}) \}$$

of base epistemic structures (Kripke models with base valuation). Define a *meta-operation*

$$\text{Lift} : U^{(k)} \longrightarrow U^{(k+1)},$$

which takes $\mathbf{E} \in U^{(k)}$ to the Σ_{ep} -structure \mathbf{E}^\uparrow with the same carrier S and relations $\{R_i\}$, and whose new predicate $\text{Sat}^{(k+1)}$ is *computed* from $\text{Sat}^{(k)}$ by the Kripke clause:

$$\text{Sat}^{(k+1)}(s, K_i \psi) \iff \forall s' (s R_i s' \Rightarrow \text{Sat}^{(k)}(s', \psi)), \quad \psi \in \mathcal{L}^{(k)}.$$

(For $k = 0$, $\text{Sat}^{(0)}$ is induced by $V^{(0)}$ and Boolean truth tables.)

Proposition 2.49 (Naturality). *If $f : (S, \{R_i\}, V^{(0)}) \xrightarrow{\cong} (S', \{R'_i\}, V'^{(0)})$ is an isomorphism of base epistemic structures, then there is a unique induced isomorphism $\text{Lift}(f) : \mathbf{E}^\uparrow \xrightarrow{\cong} (\mathbf{E}')^\uparrow$ preserving all $\text{Sat}^{(k+1)}$. Hence Lift is isomorphism-invariant.*

Proof. Isomorphisms preserve accessibility relations and base valuations. The defining clause for $\text{Sat}^{(k+1)}$ uses only R_i and $\text{Sat}^{(k)}$; the induction hypothesis gives preservation of $\text{Sat}^{(k)}$, so the displayed equivalence is preserved under f . \square

Theorem 2.50 (Representation as Iterated MetaStructure). *For every $t \in \mathbb{N}$, the tower*

$$\mathfrak{M}^{(t)} := (U^{(0)}, U^{(1)}, \dots, U^{(t)}; \text{Lift}), \quad U^{(k+1)} := \text{Lift}(U^{(k)}),$$

is an Iterated MetaStructure of depth t . Moreover, each object in $U^{(t)}$ encodes $\mathbf{MK}^{(\leq t)}(\mathcal{M})$ for some base model \mathcal{M} , and for $t = 1$ we recover ordinary Metaknowledge (Proposition 2.48).

Proof. By the Proposition, Lift is an isomorphism-invariant constructor, hence a valid meta-operation in the sense of MetaStructure. Iterating Lift adds the satisfaction predicate $\text{Sat}^{(k)}$ level by level according to Definition 2.45, thus capturing $\models^{(0)}, \dots, \models^{(t)}$ on a common carrier with fixed $\{R_i\}$. Unwinding the definitions shows that $U^{(t)}$ consists exactly of epistemic structures equipped with the first t satisfaction relations, i.e. instances of $\mathbf{MK}^{(\leq t)}(\mathcal{M})$. \square

2.7. Iterated MetaScience (Science of ... of Science)

Metascience uses scientific methods to study and improve research itself, measuring validity, reproducibility, transparency, costs, incentives, and optimizing policies systemwide (cf.[67–69]). Iterated Metascience recursively applies metascientific evaluation to policy-making processes themselves, optimizing multi-level research ecosystems through nested experimentation and loss minimization.

Definition 2.51 (Metascience as a higher-order statistical decision problem). Let Q be a measurable set of scientific questions, each with ground-truth parameter $\theta(q) \in \Theta$ and data-generating law P_q on a sample space \mathcal{X}_q . A (pre-specified) *study design* is a measurable map

$$s \in \mathcal{S} \iff E_s : \mathcal{X}_q^n \longrightarrow \mathcal{E}, \quad X \mapsto E_s(X) = (\hat{\theta}_s(X), \text{CI}_s(X), p_s(X), \delta_s(X)),$$

that turns data $X \sim P_q^n$ into an *evidential object* $E_s(X)$ (estimator, interval, p -value, decision).

An *ecosystem policy* (methods, reporting, evaluation, incentives) is denoted $\pi \in \Pi$ and induces a probability kernel \mathbb{P}_π on $Q \times \mathcal{S}$ (and thus on evidential outputs \mathcal{E}). For weights $w = (w_1, \dots, w_4) \in \mathbb{R}_{\geq 0}^4$, define per-study quality functionals

$$\text{Val}(s, q) := 1 - \text{MSE}_q(\hat{\theta}_s), \quad \text{Rep}(s, q) := \mathbb{P}\{\delta_{s'} = \delta_s \mid q, s' \sim \mathbb{P}_\pi(\cdot \mid q)\},$$

$$\text{Trn}(s) \in [0, 1] \text{ (transparency/completeness score)}, \quad \text{Cost}(s) \geq 0,$$

and the *social loss*

$$\mathcal{L}(\pi) := \mathbb{E}_{(q,s) \sim \mathbb{P}_\pi} \left[w_1(1 - \text{Val}(s, q)) + w_2(1 - \text{Rep}(s, q)) + w_3(1 - \text{Trn}(s)) + w_4 \text{Cost}(s) \right].$$

Metascience measures functionals of \mathbb{P}_π (descriptive/evaluative) and seeks $\pi^* \in \arg \min_{\pi \in \Pi} \mathcal{L}(\pi)$ (prescriptive).

Example 2.52 (MetaScience (department-level policy A/B trial)). A psychology department wants to improve the quality of senior-thesis experiments. Two ecosystem policies are considered: π_{RR} (Registered Reports + mandatory power analysis + data sharing) and π_{BAU} (business-as-usual). Over one academic year the department cluster-randomizes courses (units u) to either policy and, for each study s in unit u with question q_s , records:

$$\text{Val}(s, q_s) = 1 - \text{MSE}_{q_s}(\hat{\theta}_s),$$

$$\text{Rep}(s, q_s) = \mathbf{1}\{\delta_{s'} = \delta_s \text{ on preregistered replication}\}, \quad \text{Trn}(s) \in [0, 1], \quad \text{Cost}(s) \geq 0.$$

Using weights $w = (0.4, 0.3, 0.2, 0.1)$, the empirical social loss for policy π is

$$\widehat{\mathcal{L}}(\pi) = \frac{1}{N_\pi} \sum_{(q,s) \sim \widehat{\mathbb{P}}_\pi} \left[0.4(1 - \text{Val}) + 0.3(1 - \text{Rep}) + 0.2(1 - \text{Trn}) + 0.1 \text{Cost} \right].$$

They find $\widehat{\mathcal{L}}(\pi_{\text{RR}}) < \widehat{\mathcal{L}}(\pi_{\text{BAU}})$, driven by higher Rep and Trn with modest cost. The department adopts $\pi_{\text{RR}}^* \in \arg \min_\pi \widehat{\mathcal{L}}(\pi)$, illustrating metascience: measuring how policies π shape the distribution of evidential objects and choosing the loss-minimizing policy.

Definition 2.53 (Iterated MetaScience of depth t). A level-0 scientific ecosystem is a tuple

$$\mathbf{E}^{(0)} = (\mathcal{Q}, \{P_q\}_{q \in \mathcal{Q}}, \mathcal{S}, \Pi, \mathbb{P}_{(\cdot)}, \mathcal{L}),$$

as in the recalled definition. For $k \geq 0$, define the *lifted (meta) policy space*

$$\Pi^{(k+1)} := \Delta(\Pi^{(k)}) \quad \text{with} \quad \Pi^{(0)} := \Pi,$$

i.e. $\Pi^{(k+1)}$ is the set of randomized selectors σ over $\Pi^{(k)}$. Given $\sigma \in \Pi^{(k+1)}$, define the induced population kernel on $\mathcal{Q} \times \mathcal{S} \times \Pi^{(k)}$ by

$$\mathbb{P}_\sigma^{(k+1)} := \left[\pi \sim \sigma, (q, s) \sim \mathbb{P}_\pi^{(k)} \right], \quad \mathbb{P}_\pi^{(0)} := \mathbb{P}_\pi.$$

Define the *lifted loss* on $\Pi^{(k+1)}$ by

$$\mathcal{L}^{(k+1)}(\sigma) := \mathbb{E}_{(\pi, q, s) \sim \mathbb{P}_\sigma^{(k+1)}} \left[\mathcal{L}^{(k)}(\pi) + \lambda_{k+1} \text{Cost}^{(k+1)}(\sigma) \right], \quad \mathcal{L}^{(0)} := \mathcal{L},$$

with $\lambda_{k+1} \geq 0$ and a meta-experimental cost $\text{Cost}^{(k+1)}$. The level- $(k+1)$ meta-optimizer solves $\sigma^* \in \arg \min_{\sigma \in \Pi^{(k+1)}} \mathcal{L}^{(k+1)}(\sigma)$.

An Iterated MetaScience system of depth t is the tower

$$\mathbf{E}^{(\leq t)} := \left(\mathbf{E}^{(0)}, \Pi^{(1)}, \dots, \Pi^{(t)}; \mathbb{P}^{(1)}, \dots, \mathbb{P}^{(t)}; \mathcal{L}^{(1)}, \dots, \mathcal{L}^{(t)} \right).$$

Example 2.54 (Iterated MetaScience (funder-level experiment on policy selection rules)). A national research funder wants not only to compare policies (e.g., open-data bonuses, preregistration mandates, registered reports) but to evaluate *how* it chooses among them. Let the base policy set be $\Pi = \{\pi_1, \pi_2, \pi_3\}$ with base loss $\mathcal{L}(\pi)$ as above. Two meta-policies over Π are compared:

$$\sigma_{\text{static}} \in \Delta(\Pi) \text{ (fixed portfolio for a year),} \quad \sigma_{\text{adapt}} \text{ (Thompson-sampling bandit over } \Pi).$$

Universities are cluster-randomized to either σ_{static} or σ_{adapt} ; within each cluster, grants are assigned policies $\pi \sim \sigma$ and outcomes are tracked. The level-1 loss (Definition 2.53) is

$$\mathcal{L}^{(1)}(\sigma) = \mathbb{E}_{(\pi, q, s) \sim \mathbb{P}_\sigma^{(1)}} \left[\mathcal{L}(\pi) + \lambda_1 \text{Cost}^{(1)}(\sigma) \right],$$

where $\text{Cost}^{(1)}$ accounts for overhead of adaptivity (dashboards, monitoring). After one cycle, the funder estimates $\widehat{\mathcal{L}}^{(1)}(\sigma_{\text{adapt}}) < \widehat{\mathcal{L}}^{(1)}(\sigma_{\text{static}})$: the adaptive meta-policy quickly allocates more grants to low-loss base policies, improving reproducibility and transparency systemwide despite modest overhead. The funder then deploys $\sigma^* \in \arg \min_\sigma \widehat{\mathcal{L}}^{(1)}(\sigma)$, demonstrating *iterated* metascience: optimizing a policy-over-policies that governs how base research policies are chosen.

Proposition 2.55 (Generalization of Metascience). At depth $t = 1$, Iterated MetaScience reduces to ordinary Metascience: if σ is restricted to degenerate distributions δ_π on Π , then $\mathcal{L}^{(1)}(\delta_\pi) = \mathcal{L}^{(0)}(\pi)$ and $\arg \min_{\sigma \in \Delta(\Pi)} \mathcal{L}^{(1)}(\sigma)$ contains the embeddings of $\arg \min_{\pi \in \Pi} \mathcal{L}^{(0)}(\pi)$.

Proof. For any $\pi \in \Pi$, $\mathbb{P}_{\delta_\pi}^{(1)} = [(q, s) \sim \mathbb{P}_\pi^{(0)}]$ and hence $\mathcal{L}^{(1)}(\delta_\pi) = \mathbb{E}_{(q,s) \sim \mathbb{P}_\pi}[\dots] = \mathcal{L}^{(0)}(\pi)$, ignoring the constant experiment cost for degenerate σ . Thus minimizing $\mathcal{L}^{(1)}$ over $\Delta(\Pi)$ generalizes the base problem. \square

Fix a single-sorted signature Σ_{ms} with carrier H and function/relational symbols

$$Q, S, \text{Pi}^{(k)}, \text{Kern}^{(k)} : \text{Pi}^{(k)} \rightarrow \Delta(Q \times S \times \text{Pi}^{(k-1)}), \text{Loss}^{(k)} : \text{Pi}^{(k)} \rightarrow \mathbb{R}_{\geq 0},$$

(with the convention $\text{Pi}^{(-1)} := \emptyset$ and $\text{Kern}^{(0)} : \text{Pi}^{(0)} \rightarrow \Delta(Q \times S)$). Let $U^{(0)}$ be the class of Σ_{ms} -structures instantiating $(Q, S, \Pi, \mathbb{P}_{(\cdot)}, \mathcal{L})$.

Definition 2.56 (Meta-operation LIFT). Define $\text{Lift} : U^{(k)} \rightarrow U^{(k+1)}$ on objects by

- keeping the underlying carriers for Q and S unchanged,
- replacing $\text{Pi}^{(k)}$ by $\text{Pi}^{(k+1)} := \Delta(\text{Pi}^{(k)})$,
- defining $\text{Kern}^{(k+1)}(\sigma)$ as the mixture $\text{Kern}^{(k+1)}(\sigma) := \int \text{Kern}^{(k)}(\pi) \sigma(d\pi)$,
- defining $\text{Loss}^{(k+1)}(\sigma) := \int \text{Loss}^{(k)}(\pi) \sigma(d\pi) + \lambda_{k+1} \text{Cost}^{(k+1)}(\sigma)$.

On morphisms $f : \mathbf{E} \xrightarrow{\cong} \mathbf{E}'$ (measurable isomorphisms preserving Q, S and pushing $\text{Kern}^{(k)}, \text{Loss}^{(k)}$ forward), $\text{Lift}(f)$ acts by push-forward on $\Delta(\text{Pi}^{(k)})$.

Lemma 2.57 (Naturality of LIFT). *If $f : \mathbf{E} \xrightarrow{\cong} \mathbf{E}'$ is an isomorphism in $U^{(k)}$, then $\text{Lift}(f) : \text{Lift}(\mathbf{E}) \xrightarrow{\cong} \text{Lift}(\mathbf{E}')$ is an isomorphism in $U^{(k+1)}$ preserving $\text{Kern}^{(k+1)}$ and $\text{Loss}^{(k+1)}$.*

Proof. By assumption, f preserves $\text{Kern}^{(k)}$ and $\text{Loss}^{(k)}$ up to push-forward. For any $\sigma \in \Delta(\text{Pi}^{(k)})$,

$$\text{Kern}_{\mathbf{E}'}^{(k+1)}(\text{Lift}(f)(\sigma)) = \int \text{Kern}_{\mathbf{E}}^{(k)}(f(\pi)) \sigma(d\pi) = \int f_{\#}(\text{Kern}_{\mathbf{E}}^{(k)}(\pi)) \sigma(d\pi) = f_{\#}(\text{Kern}_{\mathbf{E}}^{(k+1)}(\sigma)),$$

and similarly $\text{Loss}^{(k+1)}$ is preserved by linearity of the integral and the invariance of $\text{Cost}^{(k+1)}$ under $\text{Lift}(f)$. Hence Lift is natural. \square

Theorem 2.58 (Iterated MetaScience is an Iterated MetaStructure). *For every $t \in \mathbb{N}$, the tower*

$$\mathfrak{M}^{(t)} := (U^{(0)}, U^{(1)}, \dots, U^{(t)}; \text{Lift}), \quad U^{(k+1)} := \text{Lift}(U^{(k)}),$$

is an Iterated MetaStructure of depth t in the sense that $U^{(k)}$ is a class of Σ_{ms} -structures and Lift is an isomorphism-invariant meta-operation (by Lemma 2.57). Moreover, any object in $U^{(t)}$ encodes an Iterated MetaScience system of depth t as in Definition 2.53, and for $t = 1$ it reduces to ordinary Metascience (Proposition 2.55).

Proof. By Lemma 2.57, LIFT is a valid meta-constructor (isomorphism-invariant). Iterating LIFT yields the sequence of policy spaces $\Pi^{(k)}$, kernels $\mathbb{P}^{(k)}$, and losses $\mathcal{L}^{(k)}$ exactly as in Definition 2.53. The identification of $U^{(t)}$ with depth- t Iterated MetaScience is immediate from the clauses in Definition 2.56. The $t=1$ case follows from Proposition 2.55. \square

2.8. Iterated MetaModel (Model of ... of Models)

A metamodel is a formal model that defines the syntax, rules, and constraints of other models, providing structure and conformance principles (cf.[70–73]). An Iterated Metamodel recursively models metamodels themselves, enabling hierarchical layers of abstraction that generalize modeling frameworks through Iterated-MetaStructure formalisms.

Definition 2.59 (Metamodel and conformance). Fix a finite set of *class symbols* Cls and *relation symbols* Rel with $\text{ar} : \text{Rel} \rightarrow \mathbb{Z}_{\geq 1}$. A *metamodel* is a tuple

$$\mathbf{MM} = (\text{Cls}, \text{Rel}, \text{ar}, \text{Ctr}),$$

where Ctr is a set of (first-order) well-formedness constraints over typed graphs. A *model* conforming to \mathbf{MM} is a finite typed multigraph

$$\mathbf{M} = (V, E, t, \ell) \quad \text{with} \quad t : V \rightarrow \text{Cls}, \ell : E \rightarrow \text{Rel},$$

such that for each $e \in E$ with $\ell(e) = r$ of arity $k = \text{ar}(r)$, the endpoints of e form an ordered k -tuple in V^k , and (V, E, t, ℓ) satisfies all constraints in Ctr . We write $\mathbf{M} \vDash \mathbf{MM}$ and call the relation \vDash *conformance*.

Example 2.60 (MetaModel in everyday software design (UML \Rightarrow domain model)). **Metamodel (UML fragment)**. The UML metamodel provides the *types of modeling elements* and their rules, e.g. *Class*, *Association*, *Attribute*, *Multiplicity*, and well-formedness constraints (OCL): every *Association* links two *Classes*; multiplicities are nonnegative; attribute types are declared, etc. Formally, it is a typed graph of meta-classes and meta-relations that any user model must *conform* to.

Model (conforming e-commerce diagram). A team models an online store with classes

Customer, Order, OrderLine, Product.

Associations (with multiplicity constraints) encode business rules:

- Customer 1-* Order (a customer can place many orders).
- Order 1-* OrderLine (each order has at least one line).
- OrderLine *-1 Product (each line references exactly one product).

Attributes include `Order.date: Date`, `OrderLine.qty: Nat`, `OrderLine.unitPrice: Money`. An OCL invariant expresses a domain constraint:

$$\text{context Order inv Total : self.total} = \sum_{i \in \text{self.lines}} (i.qty \cdot i.unitPrice).$$

This user model is valid because every element (Class, Association, Attribute, multiplicity, OCL) is an *instance* of the UML metamodel elements and satisfies their well-formedness rules. Thus the UML metamodel (the “model of models”) constrains and validates the concrete e-commerce model.

Definition 2.61 (Iterated MetaModel of order n). For $n \geq 1$, an *iterated metamodel of order n* is a finite chain

$$C_n := (\mathbf{L}_0, \mathbf{L}_1, \dots, \mathbf{L}_n)$$

with the following properties:

1. \mathbf{L}_0 is a (level-0) *model*.
2. For each $k = 1, \dots, n$, \mathbf{L}_k is a metamodel $\mathbf{L}_k = (\text{Cls}_k, \text{Rel}_k, \text{ar}_k, \text{Ctr}_k)$.
3. **Typed conformance chain:** for each $k = 1, \dots, n$ there is a conformance relation $\vDash_k \subseteq \mathcal{U}_{k-1} \times \mathcal{U}_k$ such that

$$\mathbf{L}_{k-1} \vDash_k \mathbf{L}_k,$$

where \mathcal{U}_0 is the class of level-0 models and, for $k \geq 1$, \mathcal{U}_k is the class of level- k metamodels.

We abbreviate this situation by writing a *conformance tower*

$$\mathbf{L}_0 \vDash_1 \mathbf{L}_1 \vDash_2 \dots \vDash_n \mathbf{L}_n.$$

The case $n = 1$ recovers ordinary metamodeling ($\mathbf{L}_0 \vDash_1 \mathbf{L}_1$).

Example 2.62 (Iterated MetaModel in everyday data exchange (metaschema \Rightarrow schema \Rightarrow document)). **Level M2 (metamodel-of-metamodel: JSON Schema metaschema)**. The JSON Schema *metaschema* specifies what a *schema itself* may contain: keywords like *type*, *properties*, *required*, *items*, *minimum*, and the allowed structures for combining them. It is a metamodel because its “instances” are *schemas*.

Level M1 (metamodel: an organization's ExpenseReport.schema.json). A company publishes a JSON Schema for expense reports using the metaschema's keywords:

- properties: employeeId: string, date: string (format: date), items: array.
- Each item has category $\in \{\text{travel, meals, lodging}\}$, amount: number with minimum = 0.
- required: {employeeId, date, items}.

This schema *conforms to* the metaschema (all keywords/structures are valid), so it is a correct model at M1.

Level M0 (model instance: a concrete expense report document).

An employee submits `alice_expense_0421.json`: it contains the required fields, a list of items with nonnegative amounts, and valid category values. This document *conforms to* `ExpenseReport.schema.json`.

Why this is iterated metamodeling. Conformance occurs at two successive meta-levels:

$$\underbrace{\text{document}}_{M0} \vDash \underbrace{\text{schema}}_{M1} \quad \text{and} \quad \underbrace{\text{schema}}_{M1} \vDash \underbrace{\text{metaschema}}_{M2}.$$

Thus a *metamodel* (the schema) is itself validated by a *meta-metamodel* (the metaschema), and the everyday act of submitting a JSON file is governed by this two-tier, iterated conformance chain.

To make the iteration *concrete*, we specify a level-agnostic schema \mathbf{MMM}° whose instances are precisely metamodels in the sense above.

Let

$$\text{Cls}_2 = \{\text{Class, Rel, Nat, Ctr, Var, Frm}\}.$$

Let Rel_2 contain the following relation symbols with arities indicated:

$$\text{Arg} (= \text{"argument typing"}) : \text{ar} = 3, \quad \text{Arity} : \text{ar} = 2, \quad \text{Mentions} : \text{ar} = 2, \quad \text{WellFormed} : \text{ar} = 1.$$

Intended reading: $\text{Arg}(r, j, c)$ says that for relation r the j th position has class c ; $\text{Arity}(r, k)$ sets the arity of r ; $\text{Mentions}(\varphi, \cdot)$ lists the symbols used in constraint φ ; and $\text{WellFormed}(\varphi)$ encodes syntactic well-formedness of φ as a formula over the typed graph signature.

Let Ctr_2 enforce:

- For each r there is a unique k with $\text{Arity}(r, k)$.
- For each $1 \leq j \leq k$, there is a unique c with $\text{Arg}(r, j, c)$.
- Each φ with $\text{WellFormed}(\varphi)$ only mentions declared classes/relations and is type-correct (standard first-order typing conditions).

Define

$$\mathbf{MMM}^\circ = (\text{Cls}_2, \text{Rel}_2, \text{ar}_2, \text{Ctr}_2).$$

An *encoding* Enc maps any metamodel $\mathbf{MM} = (\text{Cls}, \text{Rel}, \text{ar}, \text{Ctr})$ to a level-1 model $\text{Enc}(\mathbf{MM})$ over \mathbf{MMM}° by:

create a node v_c of type Class for each $c \in \text{Cls}$,
 a node v_r of type Rel for each $r \in \text{Rel}$,
 a node v_k of type Nat for each k in the image of ar ,
 edges witnessing $\text{Arity}(v_r, v_k)$ and $\text{Arg}(v_r, j, v_{c_j})$ for each position j ,
 and nodes v_φ of type Frm for $\varphi \in \text{Ctr}$ with Mentions edges.

One checks that $\text{Enc}(\mathbf{MM}) \vDash \mathbf{MMM}^\circ$.

Definition 2.63 (The lifting operator). The *lift* of a metamodel \mathbf{MM} is the pair

$$\text{Lift}(\mathbf{MM}) := (\mathbf{MM} \vDash_2 \mathbf{MMM}^\circ),$$

where the right component is instantiated by $\text{Enc}(\mathbf{MM})$. Inductively, for $k \geq 1$ define

$$\text{Lift}^{(k)}(\mathbf{MM}) := \mathbf{MM} \vDash_2 \underbrace{\mathbf{MMM}^\circ \vDash_3 \cdots \vDash_{k+1} \mathbf{MMM}^\circ}_{k \text{ copies}}.$$

Theorem 2.64 (Generalization of MetaModel). *Every metamodel \mathbf{MM} extends canonically to an iterated metamodel of any finite order $n \geq 1$:*

$$\text{Lift}^{(n-1)}(\mathbf{MM}) = (\mathbf{L}_0, \mathbf{L}_1, \dots, \mathbf{L}_n) \quad \text{with} \quad \mathbf{L}_1 = \mathbf{MM}, \quad \mathbf{L}_2 = \cdots = \mathbf{L}_n = \mathbf{MMM}^\circ.$$

Conversely, for any iterated metamodel C_n , the truncation $(\mathbf{L}_0, \mathbf{L}_1)$ is a metamodel with its usual conformance relation.

Proof. (Existence) We showed above that for any \mathbf{MM} the encoding $\text{Enc}(\mathbf{MM})$ conforms to \mathbf{MMM}° , i.e. $\mathbf{MM} \vDash_2 \mathbf{MMM}^\circ$. Because \mathbf{MMM}° is level-agnostic (its classes/relations describe *any* metamodel in typed-graph form), the same schema can serve as a meta-metamodel at all higher levels. Therefore the tower

$$\mathbf{L}_0 \vDash_1 \mathbf{L}_1 (= \mathbf{MM}) \vDash_2 \mathbf{L}_2 (= \mathbf{MMM}^\circ) \vDash_3 \cdots \vDash_n \mathbf{L}_n (= \mathbf{MMM}^\circ)$$

is well-defined for any $n \geq 1$.

(Converse) If $C_n = (\mathbf{L}_0, \dots, \mathbf{L}_n)$ is an iterated chain, by Definition 2.61 the pair $(\mathbf{L}_0, \mathbf{L}_1)$ already satisfies the base metamodel/conformance conditions. Hence truncation recovers an ordinary meta-modeling instance. \square

Definition 2.65 (Iterated-MetaStructure for metamodeling). An *Iterated-MetaStructure (IMS)* of height n consists of universes $\{\mathcal{U}_k\}_{k=0}^n$ and conformance relations $\{\vDash_k \subseteq \mathcal{U}_{k-1} \times \mathcal{U}_k\}_{k=1}^n$ such that:

1. \mathcal{U}_0 is the class of level-0 models; \mathcal{U}_k ($k \geq 1$) is the class of level- k metamodels.
2. (*Locality*) \vDash_k is defined by the typed-graph semantics at level k .
3. (*Chainability*) For any $x \in \mathcal{U}_{k-1}$, $y \in \mathcal{U}_k$, $z \in \mathcal{U}_{k+1}$, if $x \vDash_k y$ and $y \vDash_{k+1} z$, then x is *well-typed* at level $k+1$ via z (in particular, z validates the constraints that ensure y itself is a well-formed type system for x).

Theorem 2.66 (Representation of Iterated MetaModel inside IMS). *For each $n \geq 1$, the class of order- n iterated metamodels forms an IMS of height n with \mathcal{U}_0 the level-0 models, \mathcal{U}_1 the metamodels, and $\mathcal{U}_k = \{\mathbf{MMM}^\circ\}$ for all $k \geq 2$, and with \vDash_k the usual typed-graph conformance.*

Proof. (*Well-defined universes*) \mathcal{U}_0 and \mathcal{U}_1 are as in the base definition. By construction, \mathbf{MMM}° is a metamodel whose instances are (encodings of) metamodels; hence it can serve as the unique element of \mathcal{U}_k for $k \geq 2$.

(*Locality*) Each \vDash_k is the satisfaction of first-order constraints over typed graphs at level k , exactly as in the base case.

(*Chainability*) Suppose $\mathbf{L}_{k-1} \vDash_k \mathbf{L}_k$ and $\mathbf{L}_k \vDash_{k+1} \mathbf{L}_{k+1}$. When $k \geq 1$, $\mathbf{L}_{k+1} = \mathbf{MMM}^\circ$, whose constraints enforce that \mathbf{L}_k itself is a well-formed metamodel (unique arities, well-typed constraints, etc.). Therefore \mathbf{L}_{k-1} is typed against a *well-formed* \mathbf{L}_k , and so the composition of typings is consistent. When $k = 0$ this reduces to the ordinary “model typed by metamodel typed by meta-metamodel” situation, which holds by the same argument. Thus Definition 2.65(iii) is satisfied.

All IMS axioms hold, so the representation is established. \square

2.9. Iterated Metaoptimization (Optimization of ... of Optimizations)

Optimization is the mathematical process of selecting the best solution from feasible alternatives by minimizing or maximizing an objective function under given constraints (cf.[74–77]). Metaoptimization is the process of using one optimization algorithm to tune, configure, or control another optimization method’s parameters, improving efficiency, adaptability, and performance across diverse

problem instances [78–81]. Iterated Metaoptimization recursively applies metaoptimization over metaoptimizers themselves, creating hierarchical layers that generalize tuning, automate optimizer design, and establish Iterated-MetaStructure for adaptive optimization ecosystems.

Definition 2.67 (Meta-optimization as bilevel learning of optimizers). Let Λ be a set of *optimizer hyperparameters*, \mathcal{F} a distribution over base objectives $f : \mathcal{X} \rightarrow \mathbb{R} \cup \{+\infty\}$, and A_λ an optimization routine that, given f , returns a candidate $x_\lambda(f) \in \mathcal{X}$. Given performance and cost functionals $J(f, x)$ and $C(\lambda)$, the *meta-optimization problem* is

$$\min_{\lambda \in \Lambda} \mathbb{E}_{f \sim \mathcal{F}} [J(f, x_\lambda(f))] + \rho C(\lambda) \quad \text{subject to} \quad x_\lambda(f) \in \text{Out}(A_\lambda; f),$$

where $\rho \geq 0$ and $\text{Out}(A_\lambda; f)$ is the (possibly set-valued) output of the inner optimizer. Thus meta-optimization chooses λ to optimize *the performance of an optimizer* across problem instances.

Example 2.68 (Metaoptimization in everyday ML: tuning an optimizer for faster model training). A retail company trains a next-day demand forecaster each evening. The inner optimizer is *Adam* with hyperparameters $\lambda = (\eta, \beta_1, \beta_2, \lambda_w)$ (learning rate, momentum terms, weight decay). For a given day's objective f (training loss on that day's data), the inner routine A_λ returns weights $x_\lambda(f)$ after a fixed epoch budget. The MLOps team runs a Bayesian optimizer over λ to minimize the *expected* next-day validation loss plus training cost:

$$\min_{\lambda} \mathbb{E}_{f \sim \mathcal{F}} [J(f, x_\lambda(f))] + \rho C(\lambda),$$

where J is validation loss after training and $C(\lambda)$ captures wall-clock or GPU time. The resulting λ^* (e.g., slightly smaller η and nonzero λ_w) is then fixed for the nightly runs. This is *metaoptimization*: choosing the optimizer's hyperparameters so that the *optimizer itself* performs best across the company's rolling stream of training problems.

Definition 2.69 (Iterated metaoptimization of order n). Fix $n \geq 1$. For $k = 1, \dots, n$ let Λ_k be a (measurable) hyperparameter space at level k , with level-1 associated to the *base optimizer family* $\{A_{\lambda_1}\}_{\lambda_1 \in \Lambda_1}$. For $k \geq 2$ let a *selector* (or *compiler*) be a measurable map

$$g^{(k)} : \Lambda_k \longrightarrow \Lambda_{k-1},$$

encoding how a level- k meta-choice induces a level- $(k-1)$ hyperparameter. Given costs $C_k : \Lambda_k \rightarrow \mathbb{R}_{\geq 0}$ and weights $\rho_k \geq 0$, the *order- n iterated metaoptimization problem* is

$$\min_{\lambda_n \in \Lambda_n} \underbrace{\mathbb{E}_{f \sim \mathcal{F}} [J(f, x_{A_{\Gamma^{(n)}}(\lambda_n)}(f))] }_{\text{expected performance of the effective optimizer}} + \underbrace{\sum_{k=1}^n \rho_k C_k(\Gamma^{(k)}(\lambda_n))}_{\text{composed cost}}, \quad (\star_n)$$

where the *compiled* hyperparameters are defined by

$$\Gamma^{(1)}(\lambda_n) := g^{(2)} \circ g^{(3)} \circ \dots \circ g^{(n)}(\lambda_n) \in \Lambda_1, \quad \Gamma^{(k)}(\lambda_n) := g^{(k+1)} \circ \dots \circ g^{(n)}(\lambda_n) \in \Lambda_k$$

(with the convention that an empty composition is the identity, so $\Gamma^{(n)}(\lambda_n) = \lambda_n$), and the *effective optimizer* at order n is

$$\tilde{A}_{\lambda_n}^{(n)} := A_{\Gamma^{(1)}(\lambda_n)}.$$

We call (\star_n) an *iterated* (or multi-level) meta-optimization and (\star_1) reduces to ordinary meta-optimization.

Example 2.70 (Iterated metaoptimization in practice: tuning the tuner that tunes training). A cloud analytics platform serves many internal teams. There are three levels:

- **Level 1 (base optimizer):** for each task f , training uses SGD/Adam with hyperparameters $\lambda_1 \in \Lambda_1$; the routine A_{λ_1} returns $x_{\lambda_1}(f)$.
- **Level 2 (tuner configuration):** a selector $g^{(2)} : \Lambda_2 \rightarrow \Lambda_1$ maps a *tuner setting* λ_2 (e.g., BO kernel type, acquisition function, Hyperband parameter η , max trials) into a concrete search space and schedule for Level 1. Running the tuner with $g^{(2)}(\lambda_2)$ yields the chosen λ_1 .
- **Level 3 (portfolio policy):** a policy $g^{(3)} : \Lambda_3 \rightarrow \Lambda_2$ chooses *which* tuner (portfolio of BO, Hyperband, population-based training) and its exploration budget for a new project based on meta-features (data size, signal-to-noise, latency SLA).

When a new forecasting project arrives, the platform picks λ_3 (risk/budget preferences). This compiles to

$$\Gamma^{(1)}(\lambda_3) = g^{(2)}(g^{(3)}(\lambda_3)) \in \Lambda_1,$$

which is the effective Level-1 hyperparameter used by the training optimizer; the system minimizes

$$\min_{\lambda_3 \in \Lambda_3} \mathbb{E}_{f \sim \mathcal{F}} \left[J(f, x_{A_{\Gamma^{(1)}(\lambda_3)}}(f)) \right] + \rho_2 C_2(g^{(3)}(\lambda_3)) + \rho_3 C_3(\lambda_3).$$

In everyday terms: the platform *tunes the tuner* (Level 2) and also *tunes the policy that chooses and budgets tuners* (Level 3). A global change (e.g., stricter latency SLA) is made once at Level 3 and propagates downward, automatically altering the search strategy (Level 2) and, in turn, the training optimizer settings (Level 1) used by each team.

Definition 2.69 already presents the n -level problem as a *single* optimization over Λ_n , by compiling upper-level choices down to a level-1 optimizer through $\Gamma^{(1)}$ and accounting for all costs via the composed term.

Theorem 2.71 (Generalization and equivalence by compilation). *Let $n \geq 1$ and suppose all Λ_k are compact metric spaces, the selectors $g^{(k)}$ are continuous, C_k are lower-semicontinuous, and $f \mapsto J(f, x_{A_{\lambda_1}}(f))$ is integrably bounded with the expectation $\mathbb{E}_{f \sim \mathcal{F}}$ continuous in λ_1 . Then:*

1. ($n=1$ boundary case) For $n = 1$, (\star_n) coincides with the standard meta-optimization problem.
2. (Flattening) Any n -level meta-optimization with hierarchical decision variables $(\lambda_1, \dots, \lambda_n)$ constrained by $\lambda_{k-1} = g^{(k)}(\lambda_k)$ is equivalent to (\star_n) , i.e. to a single-level problem over λ_n with objective

$$\Phi_n(\lambda_n) = \mathbb{E}_{f \sim \mathcal{F}} \left[J(f, x_{\bar{A}_{\lambda_n}^{(n)}}(f)) \right] + \sum_{k=1}^n \rho_k C_k(\Gamma^{(k)}(\lambda_n)).$$

In particular, minimizers correspond under the bijection $\lambda_n \mapsto (\Gamma^{(1)}(\lambda_n), \dots, \Gamma^{(n)}(\lambda_n))$.

3. (Existence) Under the stated compactness/continuity assumptions, (\star_n) admits a minimizer.

Proof. (1) Immediate from the definitions: for $n = 1$ the composition $\Gamma^{(1)}$ is the identity on Λ_1 and the cost reduces to $\rho_1 C_1$.

(2) Consider the constrained hierarchical formulation

$$\min_{\lambda_1, \dots, \lambda_n} \mathbb{E}_f [J(f, x_{A_{\lambda_1}}(f))] + \sum_{k=1}^n \rho_k C_k(\lambda_k) \quad \text{s.t.} \quad \lambda_{k-1} = g^{(k)}(\lambda_k), \quad k = 2, \dots, n.$$

Successively eliminating $\lambda_{n-1}, \dots, \lambda_1$ by substitution yields the unconstrained problem

$$\min_{\lambda_n} \mathbb{E}_f [J(f, x_{A_{\Gamma^{(1)}(\lambda_n)}}(f))] + \sum_{k=1}^n \rho_k C_k(\Gamma^{(k)}(\lambda_n)),$$

which is exactly (\star_n) . The map $\lambda_n \mapsto (\lambda_1, \dots, \lambda_n) = (\Gamma^{(1)}(\lambda_n), \dots, \Gamma^{(n)}(\lambda_n))$ establishes a one-to-one correspondence between feasible tuples and choices of λ_n , and the objective values are identical by construction, hence minimizers correspond.

(3) By compactness of Λ_n and continuity of each $g^{(k)}$, each composed map $\Gamma^{(k)}$ is continuous. Lower-semicontinuity of $\sum_k \rho_k C_k \circ \Gamma^{(k)}$ follows from the corresponding property of the C_k . By the dominated convergence (or the assumed continuity of the expectation in λ_1) and continuity of $\Gamma^{(1)}$, the performance term is continuous in λ_n . Thus Φ_n is lower-semicontinuous on a compact set and attains a minimum. \square

Definition 2.72 (Iterated-MetaStructure (IMS) for metaoptimization). An IMS of height n for metaoptimization consists of universes $\{\mathcal{U}_k\}_{k=0}^n$ and relations $\{\varepsilon_k\}_{k=1}^n$ with:

1. \mathcal{U}_0 = set of base objectives with their distributions (\mathcal{F}, f, J) ;
2. \mathcal{U}_1 = set of base optimizer families $\{A_{\lambda_1}\}_{\lambda_1 \in \Lambda_1}$;
3. For $k \geq 2$, \mathcal{U}_k = set of selectors $g^{(k)} : \Lambda_k \rightarrow \Lambda_{k-1}$;
4. The relation $X \varepsilon_1 Y$ means: $Y \in \mathcal{U}_1$ acts on $X \in \mathcal{U}_0$ (produces $f \mapsto x_{A_{\lambda_1}}(f)$). For $k \geq 2$, $Y \varepsilon_k Z$ (with $Y \in \mathcal{U}_{k-1}$, $Z \in \mathcal{U}_k$) means: Z compiles a level- $(k-1)$ choice from a level- k choice, i.e. applies $g^{(k)}$.

Theorem 2.73 (Representation in IMS). For any order n , the data of an iterated metaoptimization instance

$$(\mathcal{F}, \{A_{\lambda_1}\}, \{g^{(k)}\}_{k=2}^n, \{C_k, \rho_k\}_{k=1}^n)$$

defines an IMS of height n as in Definition 2.72, and the compiled effective optimizer is the result of the iterated relation

$$\mathcal{F} \varepsilon_1 \{A_{\lambda_1}\} \varepsilon_2 g^{(2)} \varepsilon_3 \cdots \varepsilon_n g^{(n)},$$

namely $\tilde{A}_{\lambda_n}^{(n)} = A_{\Gamma^{(1)}(\lambda_n)}$. Moreover, the standard metaoptimization problem is precisely the height-1 truncation of this IMS.

Proof. By construction, ε_1 captures the action of a base optimizer on objectives. For $k \geq 2$, ε_k is the (deterministic) compilation step $\lambda_{k-1} = g^{(k)}(\lambda_k)$. Composing these relations yields $\Gamma^{(1)} = g^{(2)} \circ \cdots \circ g^{(n)}$, hence the effective optimizer $A_{\Gamma^{(1)}(\lambda_n)}$. Truncating at level 1 removes the compilers and recovers the usual metaoptimization. \square

2.10. Iterated Metaorganization (Organization of ... of Organizations)

A metaorganization is an organizational form where members are themselves organizations, coordinating collective action, governance, and decision-making across institutional boundaries [82–87]. Iterated Metaorganization recursively structures organizations of organizations, creating multi-level governance hierarchies that manage interactions, coordination, and adaptation across successive organizational layers.

Definition 2.74 (Meta-organization (MO)). Let Org be a set of organizations. Each $o \in \text{Org}$ has an interface action space \mathcal{A}_o . A meta-organization is a pair

$$\mathbf{MO} = (O, D), \quad O \subseteq \text{Org}, \quad D : \prod_{o \in O} \mathcal{A}_o \longrightarrow \mathcal{A}_{\mathbf{MO}},$$

where D is a governance/aggregation rule mapping member-organizations' action profile to a collective action in $\mathcal{A}_{\mathbf{MO}}$. (Preference, feasibility, or incentive constraints on D may be imposed but are not needed for the structural results below.)

Example 2.75 (Metaorganization in practice: a city climate coalition). A metropolitan *Climate Action Council* is formed whose members are organizations: city transit authority, utilities, chambers of commerce, and environmental NGOs. Each member o chooses an interface action $a_o \in \mathcal{A}_o$ such as an annual emission reduction target and a budget request. The council's governance rule

$$D : \prod_{o \in O} \mathcal{A}_o \longrightarrow \mathcal{A}_{\mathbf{MO}}$$

solves a transparent allocation with commitment: given reported requests $r = (r_o)_o$ and utilities u_o , pick

$$D(r) := \arg \max_{x \in \mathbb{R}_{\geq 0}^O} \sum_{o \in O} u_o(x_o) \quad \text{subject to} \quad \sum_o x_o \leq B, \quad x_o \leq r_o,$$

and publish a joint emissions plan plus funding vector x . Members remain independent organizations, but coordination and accountability occur at the meta level through D .

Definition 2.76 (Iterated metaorganization of order n). Fix a rooted finite directed tree $\mathcal{T} = (V, E)$ of depth $\leq n$. Assign to each node $v \in V$ an action space \mathcal{A}_v . Leaves $L \subseteq V$ are *atomic organizations* (no children) and have no internal rule. Every internal node v (with children $\text{ch}(v)$) is a *meta-organization node* with governance rule

$$D_v : \prod_{u \in \text{ch}(v)} \mathcal{A}_u \longrightarrow \mathcal{A}_v.$$

The triple $(\mathcal{T}, \{\mathcal{A}_v\}_{v \in V}, \{D_v\}_{v \in V \setminus L})$ is an *iterated metaorganization (IMO)*. Its *root output* is an action in the root space $\mathcal{A}_{\text{root}}$.

Example 2.77 (Iterated metaorganization: layered disaster response governance). Local NGOs and municipal agencies in several coastal cities form *city clusters* (first layer). Each cluster v aggregates its members' proposed resource deployments $a_u \in \mathcal{A}_u$ via a governance rule

$$D_v : \prod_{u \in \text{ch}(v)} \mathcal{A}_u \rightarrow \mathcal{A}_v,$$

returning a cluster deployment vector (boats, generators, medical kits). Clusters then form a *regional council* (second layer) with rule $D_{\text{reg}} : \prod_v \mathcal{A}_v \rightarrow \mathcal{A}_{\text{reg}}$ that reconciles intercluster transport and warehouse capacities. Finally, the *national task force* (third layer) applies $D_{\text{nat}} : \mathcal{A}_{\text{reg}} \rightarrow \mathcal{A}_{\text{nat}}$ to align with federal assets and no fly zones. The directed tree of organizations induces a root output in \mathcal{A}_{nat} that is an organization of organizations of organizations, enabling rapid, multi level coordination while preserving local autonomy.

Definition 2.78 (Behavior map (flattened decision rule)). For an IMO as in Definition 2.76, define recursively the *behavior map* $B_v : \prod_{\ell \in L(v)} \mathcal{A}_\ell \rightarrow \mathcal{A}_v$ for each node v , where $L(v)$ are the leaves in the subtree rooted at v :

$$B_v := \begin{cases} \text{id}_{\mathcal{A}_v} & \text{if } v \in L, \\ D_v \circ \left(\prod_{u \in \text{ch}(v)} B_u \right) & \text{if } v \notin L, \end{cases}$$

where $\prod_{u \in \text{ch}(v)} B_u$ is the canonical product map $\prod_{\ell \in L(v)} \mathcal{A}_\ell \rightarrow \prod_{u \in \text{ch}(v)} \mathcal{A}_u$. The *flattened (single-level) decision rule* of the IMO is B_{root} .

Example 2.79 (Behavior map: composing two stage budget aggregation). Consider an iterated metaorganization with leaves $L = \{A, B, C\}$ holding local budget proposals $a_A, a_B, a_C \in \mathbb{R}_{\geq 0}$. Node v_1 governs sector partners A and B by averaging their proposals

$$D_{v_1}(a_A, a_B) = \frac{1}{2}(a_A + a_B) \in \mathcal{A}_{v_1}.$$

The root node r (a national board) then combines the sector aggregate with C using a policy weight $\alpha \in (0, 1)$,

$$D_r(x, a_C) = \alpha x + (1 - \alpha) a_C \in \mathcal{A}_r.$$

The *behavior map* $B_r : \mathbb{R}_{\geq 0}^3 \rightarrow \mathbb{R}_{\geq 0}$ that flattens the tree is

$$B_r(a_A, a_B, a_C) = D_r(D_{v_1}(a_A, a_B), a_C) = \alpha \frac{a_A + a_B}{2} + (1 - \alpha) a_C,$$

a single level decision rule equivalent to the two stage governance. Stakeholders can thus audit how leaf actions deterministically propagate to the root decision.

Theorem 2.80 (IMO generalizes MO and flattens to a single MO). *Let $(\mathcal{T}, \{\mathcal{A}_v\}, \{D_v\})$ be an IMO with root r and leaf set L . Then the pair*

$$(O := L, \quad D^* := B_r : \prod_{\ell \in L} \mathcal{A}_\ell \rightarrow \mathcal{A}_r)$$

is a (single-level) meta-organization whose collective action coincides with that of the IMO for every leaf action profile. Moreover, if $\text{depth}(\mathcal{T}) = 1$, the IMO reduces to the usual MO.

Proof. By definition of B_r , for any profile $(a_\ell)_{\ell \in L} \in \prod_{\ell \in L} \mathcal{A}_\ell$, the unique action produced at the root by the hierarchical evaluation of the tree is

$$B_r((a_\ell)_{\ell \in L}) = D_r\left(B_{u_1}((a_\ell)_{\ell \in L(u_1)}), \dots, B_{u_m}((a_\ell)_{\ell \in L(u_m)})\right),$$

which, by the recursive definition of each B_{u_j} , is precisely the result of applying every internal D_v bottom-up. Hence the IMO is behaviorally equivalent to the MO with members L and rule $D^* = B_r$. If the depth is 1, the root's children are leaves, so $B_r = D_r \circ \prod_{u \in \text{ch}(r)} \text{id} = D_r$, i.e. the definition of an ordinary MO. \square

Definition 2.81 (Compositional properties). A predicate P on decision rules is *compositional* if for any arities compatible with product composition,

$$P(D_v) \text{ for all internal } v \implies P(B_r).$$

Examples: continuity (with product topologies), coordinatewise monotonicity, anonymity (when product orders/labels align), and Lipschitz boundedness, all of which are preserved under product and composition.

Proposition 2.82 (Property preservation). *If each D_v is continuous (resp. coordinatewise monotone, resp. L -Lipschitz), then so is B_r .*

Proof. Continuity and monotonicity are preserved under finite products and composition. For Lipschitzness, if each D_v has constant L_v w.r.t. sup-product norms, then B_r is Lipschitz with constant bounded by the product (or appropriate composition bound) of the L_v along each path; taking the maximum over finitely many root-to-leaf paths yields a global constant. \square

Definition 2.83 (Iterated-MetaStructure (IMS) for metaorganization). Given an IMO of depth $\leq n$, define levels \mathcal{U}_k to be the set of nodes at depth k (measured from leaves, so leaves have $k = 0$). For $k \geq 1$, define a binary relation ε_k between $\prod_{u \in \text{ch}(v)} \mathcal{A}_u$ and $v \in \mathcal{U}_k$ by

$$\left((a_u)_{u \in \text{ch}(v)} \varepsilon_k v \right) \iff D_v((a_u)_{u \in \text{ch}(v)}) \text{ is the output at } v.$$

The *iterated action* at the root is obtained by composing the relations ε_k along the unique rooted paths (i.e. relational composition mirrors functional composition).

Theorem 2.84 (Representation in IMS). *For any IMO of depth $\leq n$, the IMS in Definition 2.83 represents its evaluation in the sense that the iterated relational composition from leaves to the root yields the graph of the behavior map B_r . Consequently, ordinary meta-organization corresponds to the height-1 truncation of the IMS.*

Proof. Proceed by induction on the height h of the subtree. For $h = 1$ (a single internal node v), ε_1 is exactly the graph of D_v , hence equals the graph of B_v . Assume the claim holds for all subtrees of height $< h$. Let v have children u_1, \dots, u_m , each with height $< h$. By the induction hypothesis, the relational composition from the leaves of each subtree to u_j equals the graph of B_{u_j} . Composing these m relations (product) with ε_1 at v yields the graph of $D_v \circ (\prod_j B_{u_j})$, which is the graph of B_v by definition. Applying this at the root gives the statement. The height-1 truncation removes all intermediate compositions and leaves the single relation given by the root aggregator, the MO case. \square

2.11. Iterated Metaprogramming (Programming of ... of Programmings)

A metaprogram is a program that manipulates or generates other programs, treating code as data to transform, optimize, or produce executable structures (cf.[88,89]). An Iterated Metaprogram applies metaprogramming recursively, enabling programs that generate or transform metaprograms themselves, creating layered abstraction through Iterated-MetaStructure.

Definition 2.85 (Metaprogram). Fix a base language L_0 with abstract syntax AST_0 and semantics $\llbracket \cdot \rrbracket_0 : AST_0 \rightarrow \text{Beh}$. A (*level-1*) metaprogram is a computable transformer $P : AST_0 \rightarrow AST_0$ (or, dually, a generator $G : \Sigma^* \rightarrow AST_0$), typically expressed in a metalanguage L_1 equipped with quoting/splicing so that the denotation $\llbracket p \rrbracket_1$ of a metaterm $p \in AST_1$ satisfies $\llbracket p \rrbracket_1 = P$. A semantic contract Φ specifies correctness, e.g. $\llbracket P(e) \rrbracket_0 = \Phi(\llbracket e \rrbracket_0)$ for all $e \in AST_0$.

Example 2.86 (Metaprogramming in everyday work: generating a REST client from an API spec). A developer needs a typed client library for a shipping service. Instead of hand-writing code, they run a *metaprogram* (a code generator) that takes a machine-readable OpenAPI file `shipping.yaml` and produces executable artifacts:

- **Inputs (data about the program):** endpoints, request/response schemas, auth requirements.
- **Metaprogram action:** parse the spec, expand templates, and emit code.
- **Outputs (programs):** `ShippingClient.ts` with methods `createLabel`, `track`, etc.; type definitions; unit tests; and API docs.

Here the generator is a program that *writes other programs*. When the API changes, re-running the metaprogram regenerates consistent, type-safe client code and tests in seconds.

Definition 2.87 (Tower of metalanguages and iterated metaprogram). For $n \geq 1$ let $(L_k)_{k=0}^n$ be a stratified tower of languages with syntaxes AST_k and metalevel semantics

$$\llbracket \cdot \rrbracket_k : AST_k \longrightarrow (AST_{k-1} \rightarrow AST_{k-1}), \quad k = 1, \dots, n,$$

so each $p_k \in AST_k$ denotes a level- k transformer $P_k := \llbracket p_k \rrbracket_k$ acting on AST_{k-1} . An *iterated metaprogram of order n* is a tuple

$$\mathbf{P} = (p_n, p_{n-1}, \dots, p_1) \in AST_n \times \dots \times AST_1,$$

with *pipeline transformer*

$$\mathcal{P} := P_1 \circ P_2 \circ \dots \circ P_n : AST_0 \rightarrow AST_0.$$

Given $e \in AST_0$, its *iterated metaexpansion* is $e^{(n)} := \mathcal{P}(e) = P_1(P_2(\dots P_n(e)\dots))$.

Definition 2.88 (Contracts and compositional soundness). Suppose each P_k satisfies a contract Φ_k on behaviors: $\llbracket P_k(x) \rrbracket_{k-1} = \Phi_k(\llbracket x \rrbracket_{k-1})$ for all $x \in AST_{k-1}$. Say that (Φ_k) is *closed under composition* if $\Phi_1 \circ \dots \circ \Phi_n$ is welltyped and total on Beh.

Example 2.89 (Iterated metaprogramming in a team: a generator that writes generators). An engineering organization standardizes service scaffolding (logging, auth, CI, docs). They maintain:

1. **Level 2 (meta-generator) `OrgScaffolder`:** reads a high-level policy file (`org_policy.yaml`) and emits a *service-specific code generator* `CrudGen` configured with the organization's conventions (naming, lint rules, CI workflows, Dockerfile templates).
2. **Level 1 (generator) `CrudGen`:** takes a product team's domain schema (`orders.schema.json`) and generates an `orders-service` codebase: controllers, ORM models, migrations, OpenAPI, tests, and a GitHub Actions pipeline.
3. **Level 0 (programs):** the concrete microservice source code that ships to production.

Workflow in practice:

1. Platform team updates `org_policy.yaml` (e.g., switch to OpenTelemetry).
2. Run `OrgScaffolder` \Rightarrow regenerates `CrudGen` to include tracing hooks.

3. Product team runs the new CrudGen on their schema \Rightarrow a fresh service codebase with tracing, auth, CI, and docs appears automatically.

This is *iterated* metaprogramming: a program that creates another metaprogram, which in turn creates the final programs—allowing org-wide changes to propagate by regenerating at two levels.

Theorem 2.90 (Generalization and flattening). *Let $\mathbf{P} = (p_n, \dots, p_1)$ be an order n iterated metaprogram.*

(i) Generalization. *For $n = 1$ we recover ordinary metaprogramming: $\mathcal{P} = P_1$.*

(ii) Compositional soundness. *If each P_k satisfies Φ_k and (Φ_k) is closed under composition, then for all $e \in \text{AST}_0$,*

$$\llbracket e^{(n)} \rrbracket_0 = \llbracket \mathcal{P}(e) \rrbracket_0 = (\Phi_1 \circ \dots \circ \Phi_n)(\llbracket e \rrbracket_0).$$

(iii) Single-stage realization. *Assume the class of level-1 denotable transformers $C_1 := \{\llbracket p \rrbracket_1 : p \in \text{AST}_1\}$ is closed under composition. Then there exists $p^* \in \text{AST}_1$ with $\llbracket p^* \rrbracket_1 = \mathcal{P}$, i.e. the whole n -stage pipeline can be flattened to a single level-1 metaprogram without changing its effect on AST_0 .*

Proof. (i) is immediate from the definition with $n = 1$.

(ii) By definition of $e^{(n)}$ and the contracts,

$$\llbracket e^{(n)} \rrbracket_0 = \llbracket P_1(P_2(\dots P_n(e) \dots)) \rrbracket_0 = \Phi_1(\llbracket P_2(\dots P_n(e) \dots) \rrbracket_0).$$

Iterating the same reasoning for $k = 2, \dots, n$ yields the stated composition $(\Phi_1 \circ \dots \circ \Phi_n)(\llbracket e \rrbracket_0)$.

(iii) Since C_1 is closed under composition and $P_k \in C_1$ for each k (as each P_k is a computable transformer on AST_0 expressible at level 1 by assumption), we have $\mathcal{P} \in C_1$. Hence there exists $p^* \in \text{AST}_1$ with $\llbracket p^* \rrbracket_1 = \mathcal{P}$. \square

Definition 2.91 (Iterated-MetaStructure (IMS) presentation). Define levels $\mathcal{U}_k := \text{AST}_k$ ($k \geq 1$) and $\mathcal{U}_0 := \text{AST}_0$. For $k \geq 1$ introduce the binary relation

$$\vdash_k \subseteq \mathcal{U}_k \times (\mathcal{U}_{k-1} \times \mathcal{U}_{k-1}), \quad (p_k, x) \vdash_k y \iff y = \llbracket p_k \rrbracket_k(x).$$

Given $\mathbf{P} = (p_n, \dots, p_1)$, its *IMS trace* on $e \in \mathcal{U}_0$ is the unique $e^{(n)} \in \mathcal{U}_0$ such that

$$(p_n, e) \vdash_n e^{(n-1)}, \quad (p_{n-1}, e^{(n-1)}) \vdash_{n-1} e^{(n-2)}, \dots, (p_1, e^{(1)}) \vdash_1 e^{(0)} = e^{(n)}.$$

Equivalently, relational composition $(\vdash_1 \circ \dots \circ \vdash_n)$ is the graph of \mathcal{P} .

Theorem 2.92 (Correct IMS representation). *Let \mathbf{P} be as above. Then for all $e \in \text{AST}_0$, the iterated relational composition in the Definition yields exactly $e^{(n)} = \mathcal{P}(e)$. Moreover, the truncation to $n = 1$ recovers ordinary metaprogramming.*

Proof. By construction, $(p_k, x) \vdash_k y$ iff $y = P_k(x)$, hence the composition $(\vdash_1 \circ \dots \circ \vdash_n)$ applied to e yields $P_1(P_2(\dots P_n(e) \dots)) = \mathcal{P}(e)$. The case $n = 1$ is tautological. \square

2.12. Iterated MetaSystem (System of ... of Systems)

A MetaSystem is a higher-level construct where elements are systems themselves, coordinating, integrating, or supervising multiple subsystems into a unified framework [90]. An Iterated MetaSystem recursively organizes meta-systems over meta-systems, forming hierarchical layers that generalize system integration, coordination, and adaptation through Iterated-MetaStructure.

Definition 2.93 (Systems and configurations). Let Sys be a class of (discrete-time) dynamical systems $S = (X, U, \xi)$ with state space X , input space U , and transition map $\xi : X \times U \rightarrow X$. For $k \geq 0$ let lface_k be a set of k -ary *interfaces* (wiring/coordination blueprints). Define the configuration space

$$\text{Conf} := \bigsqcup_{k \geq 0} (\text{Sys}^k \times \text{lface}_k).$$

A *realizer* is a map $\text{Real} : \text{Conf} \rightarrow \text{Sys}$ that composes a configuration into a concrete system (e.g. by interconnection along the interface).

Definition 2.94 (Meta-operator and MetaSystem). A *meta-operator* is a function $\Psi : \text{Conf} \rightarrow \text{Conf}$ acting on systems as primitives (e.g. selection, composition, supervision, adaptation). A *MetaSystem* is a pair (\mathcal{S}, Ψ) with $\emptyset \neq \mathcal{S} \subseteq \text{Sys}$ and a meta-operator Ψ such that $\Psi(\mathcal{S}^k \times \text{Iface}_k) \subseteq \text{Conf}$ for all k . Its *realized system* on input configuration $c \in \text{Conf}$ is $\mathcal{S}^\Psi(c) := \text{Real}(\Psi(c)) \in \text{Sys}$. Intuitively, a MetaSystem is a “system about systems”: its state/data are systems, and its dynamics are Ψ over them.

Definition 2.95 (Iterated MetaSystem). Fix a tower of $n \geq 1$ meta-operators $\blacksquare = (\Psi_n, \Psi_{n-1}, \dots, \Psi_1)$ with each $\Psi_j : \text{Conf} \rightarrow \text{Conf}$. The *Iterated MetaSystem of order n* is

$$\text{IMS}_n := (\mathcal{S}, \blacksquare, \text{Real}), \quad \text{with pipeline } \mathcal{P} := \Psi_1 \circ \Psi_2 \circ \dots \circ \Psi_n : \text{Conf} \rightarrow \text{Conf}.$$

Given a configuration $c \in \text{Conf}$, its n -step meta-evolution is $c^{(n)} := \mathcal{P}(c)$ and the *realized system* is $\mathcal{S}^\blacksquare(c) := \text{Real}(c^{(n)}) \in \text{Sys}$.

Definition 2.96 (Contracts and invariants). Let \mathcal{I} be a set of system invariants (safety, stability, etc.) and $\text{Inv} \subseteq \text{Sys} \times \mathcal{I}$ a satisfaction relation. A meta-operator Ψ *respects* an invariant transformer $\Phi : \mathcal{I} \rightarrow \mathcal{I}$ if for all $c \in \text{Conf}$ and $\iota \in \mathcal{I}$,

$$(\text{Real}(c), \iota) \in \text{Inv} \implies (\text{Real}(\Psi(c)), \Phi(\iota)) \in \text{Inv}.$$

We call Φ the *contract* of Ψ . A list (Φ_n, \dots, Φ_1) is *closed under composition* if $\Phi_1 \circ \dots \circ \Phi_n$ is well defined on \mathcal{I} .

Theorem 2.97 (Generalization, compositional soundness, and flattening). Let $\text{IMS}_n = (\mathcal{S}, (\Psi_n, \dots, \Psi_1), \text{Real})$.

(i) Generalization. For $n = 1$ we recover an ordinary MetaSystem: $\mathcal{S}^\blacksquare(c) = \text{Real}(\Psi_1(c))$.

(ii) Compositional soundness. Suppose each Ψ_j respects a contract Φ_j , and (Φ_n, \dots, Φ_1) is closed under composition. Then for all $c \in \text{Conf}$ and $\iota \in \mathcal{I}$,

$$(\text{Real}(c), \iota) \in \text{Inv} \implies \left(\mathcal{S}^\blacksquare(c), (\Phi_1 \circ \dots \circ \Phi_n)(\iota) \right) \in \text{Inv}.$$

(iii) Flattening (single-stage realization). Assume the set of meta-operators $\mathcal{C} := \{\Psi : \text{Conf} \rightarrow \text{Conf}\}$ used in the platform is closed under composition. Then there exists $\Psi^* \in \mathcal{C}$ such that

$$\text{Real} \circ \Psi^* = \text{Real} \circ \Psi_1 \circ \dots \circ \Psi_n, \quad \text{i.e.} \quad \mathcal{S}^\blacksquare(c) = \text{Real}(\Psi^*(c))$$

for all $c \in \text{Conf}$. Hence any iterated metasystem is equivalent, extensionally, to a single meta-operator applied once.

Proof. (i) is immediate from the definitions with $\mathcal{P} = \Psi_1$.

(ii) Let $c^{(0)} := c$ and $c^{(j)} := \Psi_j(c^{(j-1)})$ for $j = 1, \dots, n$. By the definition of *respect*, from $(\text{Real}(c^{(0)}), \iota) \in \text{Inv}$ we get $(\text{Real}(c^{(1)}), \Phi_1(\iota)) \in \text{Inv}$, then $(\text{Real}(c^{(2)}), \Phi_2 \circ \Phi_1(\iota)) \in \text{Inv}$, and so on. By induction over j , after n steps we obtain $(\text{Real}(c^{(n)}), \Phi_1 \circ \dots \circ \Phi_n(\iota)) \in \text{Inv}$. Since $\mathcal{S}^\blacksquare(c) = \text{Real}(c^{(n)})$, the claim follows.

(iii) Closure of \mathcal{C} under composition implies $\Psi^* := \Psi_1 \circ \dots \circ \Psi_n \in \mathcal{C}$. Therefore, for all c , $\text{Real}(\Psi^*(c)) = \text{Real}(\Psi_1(\dots \Psi_n(c) \dots)) = \mathcal{S}^\blacksquare(c)$. \square

Definition 2.98 (Iterated-MetaStructure (IMS) presentation). Let the *levels* be $\mathcal{U}_0 := \text{Conf}$ and, for $k \geq 1$,

$$\vdash_k \subseteq \mathcal{C} \times \mathcal{U}_0 \times \mathcal{U}_0, \quad (\Psi, c) \vdash_k c' \iff c' = \Psi(c).$$

For a tower $\blacksquare = (\Psi_n, \dots, \Psi_1)$ and seed $c \in \mathcal{U}_0$, the IMS *trace* is $c = c^{(0)} \xrightarrow{\vdash_n} c^{(1)} \xrightarrow{\vdash_{n-1}} \dots \xrightarrow{\vdash_1} c^{(n)}$, and the realized system is $\text{Real}(c^{(n)})$.

Theorem 2.99 (Correct IMS representation). *For all towers \blacksquare and $c \in \text{Conf}$, the relational composition $(\vdash_1 \circ \dots \circ \vdash_n)$ is the graph of $\mathcal{P} = \Psi_1 \circ \dots \circ \Psi_n$, hence its unique output is $c^{(n)} = \mathcal{P}(c)$ and the realized system equals $S^\blacksquare(c)$. In particular, truncating to $n = 1$ recovers the ordinary MetaSystem step.*

Proof. By definition $(\Psi_j, c^{(j-1)}) \vdash_j c^{(j)} \iff c^{(j)} = \Psi_j(c^{(j-1)})$. Thus relational composition yields $c^{(n)} = \Psi_1(\Psi_2(\dots \Psi_n(c) \dots)) = \mathcal{P}(c)$. Applying Real gives $S^\blacksquare(c)$. \square

3. Conclusion

In this paper, we have defined various concepts related to MetaStructure and Iterated MetaStructure. In the future, we expect to explore algorithmic research on these structures and consider possible extensions using Fuzzy Sets [91–93], Intuitionistic Fuzzy Sets [94,95], Neutrosophic Sets [96–99], Rough Sets [100–102], HyperFuzzy Sets [28,103–106], and Plithogenic Sets [9,20,107–111].

Supplementary Materials: No supplementary materials accompany this paper.

Funding: No external funding was received for this work.

Data Availability Statement: This paper is theoretical and did not generate or analyze any empirical data. We welcome future studies that apply and test these concepts in practical settings.

Acknowledgments: We thank all colleagues, reviewers, and readers whose comments and questions have greatly improved this manuscript. We are also grateful to the authors of the works cited herein for providing the theoretical foundations that underpin our study. Finally, we appreciate the institutional and technical support that enabled this research.

Conflicts of Interest: The authors declare no conflicts of interest regarding the publication of this work.

Research Integrity: The author confirms that this manuscript is original, has not been published elsewhere, and is not under consideration by any other journal.

Use of Computational Tools: All proofs and derivations were performed manually; no computational software (e.g., Mathematica, SageMath, Coq) was used.

Code Availability: No code or software was developed for this study.

Ethical Approval: This research did not involve human participants or animals, and therefore did not require ethical approval.

Use of Generative AI and AI-Assisted Tools: We use generative AI and AI-assisted tools for tasks such as English grammar checking, and We do not employ them in any way that violates ethical standards.

References

1. Fujita, T.; Smarandache, F. A Unified Framework for U -Structures and Functorial Structure: Managing Super, Hyper, SuperHyper, Tree, and Forest Uncertain Over/Under/Off Models. *Neutrosophic Sets and Systems* **2025**, *91*, 337–380.
2. Fujita, T. How to Represent $A \rightarrow B \rightarrow \dots \rightarrow Z$: From Curried Functions and Hyperfunctions to Curried Structures and Hyperstructures, and More, 2025. <https://doi.org/10.13140/RG.2.2.20554.48323>.
3. Hausdorff, F. *Set theory*; Vol. 119, American Mathematical Soc., 2021.
4. Jech, T. *Set theory: The third millennium edition, revised and expanded*; Springer, 2003.
5. Panti, G. Multi-valued logics. In *Quantified representation of uncertainty and imprecision*; Springer, 1998; pp. 25–74.
6. Jaynes, E.T.; Grandy, T.B.; Smith, R.; Lored, T.; Tribus, M.; Skilling, J.; Bretthorst, G.L. Probability theory: the logic of science. *The Mathematical Intelligencer* **2005**, *27*, 83.
7. Kingman, J.F.C.; Feller, W. An Introduction to Probability Theory and its Applications. *Biometrika* **1958**, *130*, 430–430.
8. Chen, J.; Ye, J.; Du, S. Scale Effect and Anisotropy Analyzed for Neutrosophic Numbers of Rock Joint Roughness Coefficient Based on Neutrosophic Statistics. *Symmetry* **2017**, *9*, 208.
9. Smarandache, F. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948* **2018**.

10. Lenz, R.; Bovik, A.C. Group Theory. 2007.
11. Burde, D. Group Theory. *Computers, Rigidity, and Moduli* **2019**.
12. Wisbauer, R. *Foundations of module and ring theory*; Routledge, 2018.
13. Stenstrom, B. *Rings of quotients: an introduction to methods of ring theory*; Vol. 217, Springer Science & Business Media, 2012.
14. Dehghan, O.; Ameri, R.; Aliabadi, H.E. Some results on hypervector spaces. *Italian Journal of Pure and Applied Mathematics* **2019**, *41*, 23–41.
15. Sameena, K.; et al. FUZZY MATROIDS FROM FUZZY VECTOR SPACES. *South East Asian Journal of Mathematics and Mathematical Sciences* **2021**, *17*, 381–390.
16. Hatip, A.; Olgun, N.; et al. On the Concepts of Two-Fold Fuzzy Vector Spaces and Algebraic Modules. *Journal of Neutrosophic and Fuzzy Systems* **2023**, *7*, 46–52.
17. Diestel, R. Graph theory 3rd ed. *Graduate texts in mathematics* **2005**, *173*, 12.
18. Gross, J.L.; Yellen, J.; Anderson, M. *Graph theory and its applications*; Chapman and Hall/CRC, 2018.
19. Diestel, R. *Graph theory*; Springer (print edition); Reinhard Diestel (eBooks), 2024.
20. Smarandache, F. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neuro-/Anti-) HyperAlgebra*; Infinite Study, 2020.
21. Hou, Z. Automata Theory and Formal Languages. *Texts in Computer Science* **2021**.
22. Bonakdarpour, B.; Sheinvald, S. Automata for hyperlanguages. *arXiv preprint arXiv:2002.09877* **2020**.
23. Kovach, N.; Gibson, A.S.; Lamont, G.B. Hypergame Theory: A Model for Conflict, Misperception, and Deception. 2015.
24. House, J.T.; Cybenko, G.V. Hypergame theory applied to cyber attack and defense. In Proceedings of the Defense + Commercial Sensing, 2010.
25. Oguz, G.; Davvaz, B. Soft topological hyperstructure. *J. Intell. Fuzzy Syst.* **2021**, *40*, 8755–8764.
26. Vougioukli, S. Helix hyperoperation in teaching research. *Science & Philosophy* **2020**, *8*, 157–163.
27. Vougioukli, S. HELIX-HYPEROPERATIONS ON LIE-SANTILLI ADMISSIBILITY. *Algebras Groups and Geometries* **2023**.
28. Fujita, T. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*; Biblio Publishing, 2025.
29. Smarandache, F. Foundation of SuperHyperStructure & Neutrosophic SuperHyperStructure. *Neutrosophic Sets and Systems* **2024**, *63*, 21.
30. Smarandache, F. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*; Infinite Study, 2023.
31. Das, A.K.; Das, R.; Das, S.; Debnath, B.K.; Granados, C.; Shil, B.; Das, R. A Comprehensive Study of Neutrosophic SuperHyper BCI-Semigroups and their Algebraic Significance. *Transactions on Fuzzy Sets and Systems* **2025**, *8*, 80.
32. Fujita, T. Chemical Hyperstructures, SuperHyperstructures, and SHv-Structures: Toward a Generalized Framework for Hierarchical Chemical Modeling. *ChemRxiv* **2025**. <https://doi.org/10.26434/chemrxiv-2025-zk8pm>.
33. Fujita, T. MetaStructure, Meta-HyperStructure, and Meta-SuperHyperStructure, 2025. Preprint, <https://doi.org/10.13140/RG.2.2.22609.03680>.
34. Fujita, T. MetaHyperGraphs, MetaSuperHyperGraphs, and Iterated MetaGraphs: Modeling Graphs of Graphs, Hypergraphs of Hypergraphs, Superhypergraphs of Superhypergraphs, and Beyond, 2025. <https://doi.org/10.31224/5103>.
35. Fujita, T. Meta-Fuzzy Graph, Meta-Neutrosophic Graph, Meta-Digraph, and Meta-MultiGraph with some applications **2025**.
36. Fujita, T. MetaFuzzy, MetaNeutrosophic, MetaSoft, and MetaRough Set **2025**.
37. Tanner, K.D. Promoting student metacognition. *CBE-Life Sciences Education* **2012**, *11*, 113–120.
38. Cox, M.T. Metacognition in computation: A selected research review. *Artificial intelligence* **2005**, *169*, 104–141.
39. Schneider, W.; Artelt, C. Metacognition and mathematics education. *ZDM* **2010**, *42*, 149–161.
40. Gourgey, A.F. Metacognition in basic skills instruction. *Instructional science* **1998**, *26*, 81–96.
41. Feurer, E.; Sassu, R.; Cimeli, P.; Roebbers, C.M. Development of meta-representations: Procedural metacognition and the relationship to theory of mind. *Journal of Educational and Developmental Psychology* **2015**, *5*, 6.
42. Meyer*, J.H.; Shanahan.; P, M. Developing metalearning capacity in students: Actionable theory and practical lessons learned in first-year economics. *Innovations in Education and Teaching International* **2004**, *41*, 443–458.

43. Losada, M. The complex dynamics of high performance teams. *Mathematical and computer modelling* **1999**, *30*, 179–192.
44. Rutter, C.M.; Gatsonis, C.A. A hierarchical regression approach to meta-analysis of diagnostic test accuracy evaluations. *Statistics in medicine* **2001**, *20*, 2865–2884.
45. Senn, S.; Gavini, F.; Magrez, D.; Scheen, A. Issues in performing a network meta-analysis. *Statistical Methods in Medical Research* **2013**, *22*, 169–189.
46. Van Valkenhoef, G.; Lu, G.; De Brock, B.; Hillege, H.; Ades, A.; Welton, N.J. Automating network meta-analysis. *Research synthesis methods* **2012**, *3*, 285–299.
47. Zhao, J.; van Valkenhoef, G.; de Brock, E.; Hillege, H. ADDIS: an automated way to do network meta-analysis **2012**.
48. Hyland, K. Metadiscourse: What is it and where is it going? *Journal of pragmatics* **2017**, *113*, 16–29.
49. Jiang, F.K.; Akbaş, E. Metadiscourse in a disciplinary context: An overview. *Chinese Journal of Applied Linguistics* **2024**, *47*, 163–177.
50. Zhu, Z.; Wu, X. Metadiscourse in Multimodal Discourse: The Case of About-us Pages of Chinese and American Companies. *Journal of Modern Research in English Language Studies* **2025**, *12*, 1–18.
51. Hyland, K.; Wang, W.; Jiang, F.K. Metadiscourse across languages and genres: An overview. *Lingua* **2022**, *265*, 103205.
52. Hyland, K. Metadiscourse. In *Conducting genre-based research in applied linguistics*; Routledge, 2023; pp. 59–81.
53. Lee, K. *Philosophy and revolutions in genetics: Deep science and deep technology*; Springer, 2016.
54. Koskela, L.; et al. *Application of the new production philosophy to construction*; Vol. 72, Stanford university Stanford, 1992.
55. Teichman, J. *The Mind and the Soul: an Introduction to the Philosophy of Mind*; Routledge, 2014.
56. Smarandache, F. A unifying field in Logics: Neutrosophic Logic. In *Philosophy*; American Research Press, 1999; pp. 1–141.
57. Overgaard, S.; Gilbert, P.; Burwood, S. *An introduction to metaphilosophy*; Cambridge University Press, 2013.
58. Joll, N. *Metaphilosophy* **2010**.
59. Vasilyev, V.V. Metaphilosophy: History and Perspectives. *Epistemology & Philosophy of Science* **2019**, *56*, 6–18.
60. Schmid, J. *The methods of metaphilosophy*; Klostermann, 2022.
61. Evans, J.A.; Foster, J.G. Metaknowledge. *Science* **2011**, *331*, 721–725.
62. Avenali, A.; Daraio, C.; Di Leo, S.; Matteucci, G.; Nepomuceno, T. Systematic reviews as a metaknowledge tool: caveats and a review of available options. *International Transactions in Operational Research* **2023**, *30*, 2761–2806.
63. Trinquart, L.; Johns, D.M.; Galea, S. Why do we think we know what we know? A metaknowledge analysis of the salt controversy. *International Journal of Epidemiology* **2016**, *45*, 251–260.
64. Han, Y.; Dunning, D. Metaknowledge of experts versus nonexperts: do experts know better what they do and do not know? *Journal of Behavioral Decision Making* **2024**, *37*, e2375.
65. Pinski, M.; Haas, M.J.; Benlian, A. Building metaknowledge in ai literacy—the effect of gamified vs. text-based learning on ai literacy metaknowledge **2024**.
66. Zhang, C.; Guan, J. How to identify metaknowledge trends and features in a certain research field? Evidences from innovation and entrepreneurial ecosystem. *Scientometrics* **2017**, *113*, 1177–1197.
67. Elson, M.; Huff, M.; Utz, S. Metascience on peer review: Testing the effects of a study's originality and statistical significance in a field experiment. *Advances in Methods and Practices in Psychological Science* **2020**, *3*, 53–65.
68. Munafò, M. *Metascience: reproducibility blues*, 2017.
69. Ioannidis, J.P.; Fanelli, D.; Dunne, D.D.; Goodman, S.N. Meta-research: evaluation and improvement of research methods and practices. *PLoS biology* **2015**, *13*, e1002264.
70. Mohanty, S. Chapter 12 Metamodel-Based Fast AMS-SoC Design Methodologies," *Nanoelectronic Mixed-Signal System Design*", ISBN 978-0071825719 and 0071825711, 2015.
71. Van Gigch, J.P. *System design modeling and metamodeling*; Springer Science & Business Media, 1991.
72. Gonzalez-Perez, C.; Henderson-Sellers, B. *Metamodelling for software engineering*; John Wiley & Sons, 2008.
73. Fatehah, M.; Mezhuyev, V.; Al-Emran, M. A systematic review of metamodelling in software engineering. *Recent Advances in Intelligent Systems and Smart Applications* **2021**, pp. 3–27.
74. Koziel, S.; Yang, X.S. *Computational optimization, methods and algorithms*; Vol. 356, Springer, 2011.
75. Yi, D.; Ahn, J.; Ji, S. An effective optimization method for machine learning based on ADAM. *Applied Sciences* **2020**, *10*, 1073.

76. Nassif, N.; Al-Sadoon, Z.A.; Hamad, K.; Altoubat, S. Cost-based optimization of shear capacity in fiber reinforced concrete beams using machine learning. *Struct Eng Mech* **2022**, *83*, 671–680.
77. Gaspar-Cunha, A.; Costa, P.; Monaco, F.; Delbem, A. Many-objectives optimization: a machine learning approach for reducing the number of objectives. *Mathematical and Computational Applications* **2023**, *28*, 17.
78. Krus, P.; Ölvander, J. Performance index and meta-optimization of a direct search optimization method. *Engineering optimization* **2013**, *45*, 1167–1185.
79. Vințan, L.; Chiș, R.; Ismail, M.A.; Coțofană, C. Improving computing systems automatic multiobjective optimization through meta-optimization. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2015**, *35*, 1125–1129.
80. Mason, K.; Duggan, J.; Howley, E. A meta optimisation analysis of particle swarm optimisation velocity update equations for watershed management learning. *Applied Soft Computing* **2018**, *62*, 148–161.
81. Bian, W.; Hu, W. Balancing the Performance-Efficiency Trade-off in Lighting Control Systems through Meta-Optimisation. In Proceedings of the 2024 IEEE Sustainable Smart Lighting World Conference & Expo (LS24). IEEE, 2024, pp. 1–4.
82. Berkowitz, H.; Dumez, H. The concept of meta-organization: Issues for management studies. *European Management Review* **2016**, *13*, 149–156.
83. Coulombel, P.; Berkowitz, H. One name for two concepts: A systematic literature review about meta-organizations. *International Journal of Management Reviews* **2025**, *27*, 151–173.
84. Berkowitz, H.; Bor, S. Why meta-organizations matter: A response to Lawton et al. and Spillman. *Journal of Management Inquiry* **2018**, *27*, 204–211.
85. Gulati, R.; Puranam, P.; Tushman, M. Meta-organization design: Rethinking design in interorganizational and community contexts. *Strategic management journal* **2012**, *33*, 571–586.
86. Marciniak, R. From organization design to meta organization design. In Proceedings of the Digital Enterprise Design and Management 2013: Proceedings of the First International Conference on Digital Enterprise Design and Management DED&M 2013. Springer, 2013, pp. 147–159.
87. Napolitano, P.; Cerveró-Romero, F. Meta-Organization: the Future for the Lean Organization'. In Proceedings of the 20th Annual Conference of the International Group for Lean Construction. San Diego, USA, 2012, pp. 18–20.
88. Sobolewski, M. Exerted enterprise computing: from protocol-oriented networking to exertion-oriented networking. In Proceedings of the OTM Confederated International Conferences "On the Move to Meaningful Internet Systems". Springer, 2010, pp. 182–201.
89. Sobolewski, M. Technology foundations. In *Concurrent Engineering in the 21st Century: Foundations, Developments and Challenges*; Springer, 2015; pp. 67–99.
90. Palmer, K.D. 9.6 Meta-systems Engineering: A New Approach to Systems Engineering based on Emergent Meta-Systems and Holonomic Special Systems Theory. In Proceedings of the INCOSE International Symposium. Wiley Online Library, 2000, Vol. 10, pp. 889–904.
91. Zadeh, L.A. Fuzzy sets. *Information and control* **1965**, *8*, 338–353.
92. Al-Hawary, T. Complete fuzzy graphs. *International Journal of Mathematical Combinatorics* **2011**, *4*, 26.
93. Mordeson, J.N.; Nair, P.S. *Fuzzy graphs and fuzzy hypergraphs*; Vol. 46, Physica, 2012.
94. Atanassov, K.T.; Gargov, G. *Intuitionistic fuzzy logics*; Springer, 2017.
95. Atanassov, K.T. Circular intuitionistic fuzzy sets. *Journal of Intelligent & Fuzzy Systems* **2020**, *39*, 5981–5986.
96. Smarandache, F. *Neutrosophic Overset, Neutrosophic Underset, and Neutrosophic Offset. Similarly for Neutrosophic Over-/Under-/Off-Logic, Probability, and Statistics*; Infinite Study, 2016.
97. Zhu, S. Neutrosophic n-SuperHyperNetwork: A New Approach for Evaluating Short Video Communication Effectiveness in Media Convergence. *Neutrosophic Sets and Systems* **2025**, *85*, 1004–1017.
98. Broumi, S.; Talea, M.; Bakali, A.; Smarandache, F. Single valued neutrosophic graphs. *Journal of New theory* **2016**, pp. 86–101.
99. Broumi, S.; Talea, M.; Bakali, A.; Smarandache, F. Interval valued neutrosophic graphs. *Critical Review, XII* **2016**, *2016*, 5–33.
100. Pawlak, Z. Rough sets. *International journal of computer & information sciences* **1982**, *11*, 341–356.
101. Broumi, S.; Smarandache, F.; Dhar, M. Rough neutrosophic sets. *Infinite Study* **2014**, *32*, 493–502.
102. Pawlak, Z. *Rough sets: Theoretical aspects of reasoning about data*; Vol. 9, Springer Science & Business Media, 2012.
103. Jun, Y.B.; Hur, K.; Lee, K.J. Hyperfuzzy subalgebras of BCK/BCI-algebras. *Annals of Fuzzy Mathematics and Informatics* **2017**.

104. Ghosh, J.; Samanta, T.K. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol* **2012**, *41*, 27–37.
105. Jun, Y.B.; Song, S.Z.; Kim, S.J. Length-fuzzy subalgebras in BCK/BCI-algebras. *Mathematics* **2018**, *6*, 11.
106. Jun, Y.B.; Song, S.Z.; Kim, S.J. Distances between hyper structures and length fuzzy ideals of BCK/BCI-algebras based on hyper structures. *Journal of Intelligent & Fuzzy Systems* **2018**, *35*, 2257–2268.
107. Sultana, F.; Gulistan, M.; Ali, M.; Yaqoob, N.; Khan, M.; Rashid, T.; Ahmed, T. A study of plithogenic graphs: applications in spreading coronavirus disease (COVID-19) globally. *Journal of ambient intelligence and humanized computing* **2023**, *14*, 13139–13159.
108. Singh, P.K. Complex plithogenic set. *International Journal of Neutrosophic Sciences* **2022**, *18*, 57–72.
109. Smarandache, F. Plithogeny, plithogenic set, logic, probability and statistics: a short review. *Journal of Computational and Cognitive Engineering* **2022**, *1*, 47–50.
110. Smarandache, F. *Plithogeny, plithogenic set, logic, probability, and statistics*; Infinite Study, 2017.
111. Kandasamy, W.V.; Ilanthenral, K.; Smarandache, F. *Plithogenic Graphs*; Infinite Study, 2020.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.