The Attralucian Essays: Exploring the Finite

First Edition

Typeset in LaTeX

The Attralucian Essays

Finite Overlap and Convolution: A Finite Symbolic Mechanics Treatment

Kevin R. Haylett

Chapter 1

Finite Overlap and Convolution: A Finite Symbolic Mechanics Treatment

Overview

Convolution, cross-correlation, and autocorrelation are typically presented as algebraic or integral operations defined over functions or sequences. These formulations compress an underlying sequential process involving displacement, interaction, and accumulation. This essay makes that implicit process explicit. We define a *finite overlap operator* that captures the essential structure: a bounded index set, a displacement parameter, a local interaction function, and accumulation over ordered steps. Classical convolution and its relatives are recovered as special cases of this operator. The aim is to supplement classical formulations with a process-level description useful for computational interpretation, resource estimation, generalisation to non-multiplicative interactions, and analysis of symbolic overlap in high-dimensional systems such as language models.

Classical Formulation

Let f and g be functions defined over $\mathbb{R}$ or sequences defined over $\mathbb{Z}$.

Continuous Convolution

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)\, g(t - \tau)\, d\tau$$

Discrete Convolution

For sequences $f, g : \mathbb{Z} \to \mathbb{R}$:

$$(f * g)(n) = \sum_{k \in \mathbb{Z}} f(k)\, g(n - k)$$

In any actual computation — and within the present framework — the sum is taken over a **finite index set** $K \subset \mathbb{Z}$, either because the sequences have finite support or because the domain is truncated at a practically sufficient bound.

Cross-Correlation

$$(f \star g)(n) = \sum_{k \in \mathbb{Z}} f(k)\, g(k+n)$$

Measures similarity between f and shifted versions of g without reversal.

Autocorrelation

$$R_f(n) = \sum_{k \in \mathbb{Z}} f(k)\, f(k+n)$$

Measures self-similarity under displacement.

Hidden Sequential Structure

Although presented as static expressions, these operations imply a sequence of steps:

1. A domain is discretised or sampled.
2. One function is displaced relative to the other.
3. At each displacement, local interactions are computed.
4. These interactions are accumulated to produce an output value.

Thus, each output index n corresponds to a finite process involving:

- Storage of intermediate values
- Ordered traversal of indices
- Repeated evaluation of pairwise interactions

The integral or summation notation compresses this process into a single symbolic form. Any actual computation — whether in hardware, software, or mathematical approximation — must instantiate these steps explicitly.

Finite Overlap Operator

Define a general finite overlap operator:

$$\mathcal{O}(f, g; \delta) = \sum_{k \in K} I\big(f(k), g(k - \delta)\big)$$

where:

- f, g are finite sequences or sampled functions
- $\delta \in \mathbb{Z}$ is a displacement parameter
- $K \subset \mathbb{Z}$ is a **finite** index set
- $I : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is a local interaction function

This operator explicitly encodes:

Component	Role
δ	Displacement between sequences
I	Local interaction (not necessarily multiplication)
K	Bounded domain of summation
$\sum$	Accumulation over ordered steps

The finiteness of K is essential: any realisable computation operates over a bounded set of indices.

Recovery of Classical Cases

Classical operations are recovered as special cases. Let $\tilde{g}(k) = g(-k)$ (time reversal). Then:

$$(f * g)(n) = \mathcal{O}(f, \tilde{g}; n) \quad \text{with} \quad I(x, y) = x \cdot y$$

$$(f \star g)(n) = \mathcal{O}(f, g; -n) \quad \text{with} \quad I(x, y) = x \cdot y$$

$$R_f(n) = \mathcal{O}(f, f; -n) \quad \text{with} \quad I(x, y) = x \cdot y$$

Operation	Classical form (idealised)	Finite overlap fo
Convolution	$\sum_{k\in\mathbb{Z}} f(k)g(n-k)$	$\mathcal{O}(f, \tilde{g}; n)$ with $I(x, y$
Cross-correlation	$\sum_{k\in\mathbb{Z}} f(k)g(k+n)$	$\mathcal{O}(f, g; -n)$ with $I(x,$
Autocorrelation	$\sum_{k\in\mathbb{Z}} f(k)f(k+n)$	$\mathcal{O}(f, f; -n)$ with $I(x,$

In the classical forms shown above, summation over $\mathbb{Z}$ is an idealisation. For any finite computation — and within the present framework — the sum is taken over a **finite index set** $K \subset \mathbb{Z}$ that is practically sufficient for the domain. The finite overlap operator $\mathcal{O}$ makes this bounded domain explicit through its parameter K.

Beyond Multiplication: Generalised Interaction

The finite overlap operator allows I to be any measurable interaction, not just multiplication. Examples include:

Interaction $I(x, y)$	Interpretation
$x \cdot y$	Classical linear convolution
$\max(0, x + y)$	Thresholded additive overlap
$\mathbf{1}[x = y]$	Symbolic equality (set intersection)
$\exp(-\|x - y\|^2)$	Similarity kernel
$\|x - y\|^2$	Squared distance (displacement cost)

This generality makes the operator useful beyond signal processing — for example, in comparing symbolic sequences or measuring structural overlap in representational spaces.

Temporal Unfolding Interpretation

Define:

$$C(n) = \mathcal{O}(f, g; n)$$

Then $C(n)$ is not merely a value, but the result of a finite sequence of operations indexed by n.

The sequence $\{C(n)\}_{n\in\mathbb{Z}}$ may be interpreted as a **trajectory of interaction states**. Each step in n corresponds to a new configuration of overlap between f and g.

This reframes convolution as a process evolving over discrete states, rather than a static functional mapping. The unfolding consists of making this sequential dependence explicit.

> **Example:** For $n = -2, -1, 0, 1, 2$, each $C(n)$ requires a fresh traversal of $k \in K$. The total operation count is $|K| \cdot |N|$ pairwise interactions, where N is the set of output indices.

Multi-Basin Interaction

Let $\{T_i\}_{i=1}^{m}$ be a finite family of sequences (treatments, templates, or basis functions), and let X be an input sequence.

Define:

$$Y(n) = \sum_{i=1}^{m} w_i \sum_{k\in K} I\big(T_i(k), X(n-k)\big)$$

where $w_i \in \mathbb{R}$ are weights.

Interpretation:

- Each T_i acts as an interaction template
- w_i controls its contribution
- The result $Y(n)$ reflects combined interaction across multiple structures

This can be seen as a **finite superposition of overlap processes** — a form of multi-template matching or attention-like pooling.

Application: Symbolic Overlap in High-Dimensional Systems

In high-dimensional systems such as language models, symbolic sequences (sentences, documents) may be mapped into vector representations $V_A, V_B \subset \mathbb{R}^d$ (e.g., activation vectors across layers or tokens).

Define a **projected overlap**:

$$R(A, B) = \Pi\left(V_A \cap V_B\right)$$

where:

- $V_A \cap V_B$ represents overlap in representation space (e.g., shared directions or clusters)

- Π is a projection operator into a measurable low-dimensional subspace

This does not capture meaning in the human sense, but provides a **measurable estimate of structural interaction** under a given model.

Conjecture of Resonance

Let two texts A and B be embedded into an LLM's activation space. Define the resonance volume:

$$\mathrm{Res}(A, B) = \sum_{\ell=1}^{L} \sum_{k \in K_\ell} I\left(h_\ell^{(A)}(k), h_\ell^{(B)}(k-\delta)\right)$$

across layers ℓ, positions k, and a suitable interaction I (e.g., cosine similarity). Then:

- High resonance implies high structural overlap in the model's internal representations
- Resonance profiles are **invariant** under meaning-preserving paraphrases (empirical prediction)
- Resonance profiles change under permutations or structural perturbations

This yields a **testable hypothesis** for LLM interpretability: symbolic overlap can be measured without reconstructing semantics.

Discussion

The classical formulations of convolution, cross-correlation, and autocorrelation are typically treated as algebraic or analytical objects. However, in practical computation, these expressions are always realised through finite procedures:

- Integrals are approximated by discrete sums
- Domains are truncated at practically sufficient bounds
- Functions are sampled at finite resolution
- Operations are performed sequentially

Thus, even within classical mathematics, the symbolic expressions are operationally reduced to finite processes. The finite overlap operator makes this reduction explicit, providing a bridge between:

Domain	Description
Classical analysis	Idealised limits, unbounded domains
Computational mathematics	Discrete approximations, bounded sup
Symbolic dynamics	Ordered state transitions, sequential o

Within this reframing:

- Symbols arise from measurable distinctions
- Operations correspond to ordered transformations
- Equations represent compressed traces of these trans-

formations

The finite overlap operator provides a general mechanism for expressing interaction between symbolic structures. Classical convolution is understood as a specific instance within this broader class.

Conclusion

Convolution and related operations may be understood as compressed representations of finite overlap processes. By making their sequential structure explicit, we reveal displacement, interaction, and accumulation as fundamental components.

The finite overlap operator

$$\mathcal{O}(f, g; \delta) = \sum_{k \in K} I(f(k), g(k - \delta))$$

generalises classical convolution to:

- Bounded, finite index sets
- Arbitrary interaction functions
- Explicit displacement ordering

This reframing does not replace classical analysis. It supplements it with a process-level description, useful for computational interpretation, resource estimation, generalisation to non-multiplicative interactions, and analysing symbolic overlap in high-dimensional systems such as language models.

The symbolic form is not primary. It is the compressed trace of a finite ordered process.

Afterword: Boundary Conditions and Further Unfolding

The preceding essay has remained within what might be called the *classical basin* of discourse. It uses conventional mathematical language, appeals to computation where necessary, and makes finiteness explicit without invoking a broader philosophical frame. This is by design: the finite overlap operator stands on its own as a mathematical object useful for signal processing, symbolic dynamics, and LLM interpretability.

However, the attentive reader may have noticed certain points where the presentation touches deeper waters:

- The notion of a "practically sufficient bound" for the index set K
- The distinction between summation over $\mathbb{Z}$ as an idealisation versus summation over a finite K as a computation
- The claim that infinity is not attained but approached as a temporal process

- The interpretation of equations as compressed traces of procedures rather than static relations

These are not incidental. They point toward a more foundational framework called **Finite Symbolic Mechanics (FSM)** and its associated method, **Finite Process Unfolding (FPU)**.

Boundary Conditions

In FSM, every symbolic operation occurs within a bounded container. The boundary of this container is not arbitrary: it corresponds to the *generonic sphere* — the finest measurement distinguishable at a given epoch. Within this sphere, symbols are produced directly by *exogenous* generonic events (measurements at the boundary). Beyond it, structure is *inferred* via *endogenous* generonic events (modelling within the symbolic space).

The finite overlap operator $\mathcal{O}$ as presented above is, in FSM terms, an *exogenous* operator when K is bounded by the generonic sphere. The classical convolution integral, by contrast, is an *endogenous* projection — an inference that extrapolates from measured finite overlaps to an idealised continuum.

Endogenous and Exogenous Measurement

The distinction between exogenous and endogenous measurement resolves a longstanding tension: how can a finite symbolic system represent infinite or continuous structures? The answer is that it does not measure them directly. It models them endogenously from boundary measurements. The classical integral is not primary; it is a stable compression of a finite procedure followed by an admissible inference.

Toward Finite Symbolic Mathematics

The framework sketched here — Finite Symbolic Mechanics — treats mathematics not as a description of a pre-existing ideal realm, but as a nonlinear dynamical system of symbolic trajectories. Equations are stable compressions of finite processes. Infinity is replaced by practical computable limits. Distinguishability has a cost. And every symbolic expression carries the trace of its own generation.

This afterword is not the place for a full exposition of FSM. Interested readers are directed to the *Attralucian Essays*, where the axioms of Finite Symbolic Mechanics, the method of Finite Process Unfolding, and the distinction between exogenous and endogenous generons are developed in full.

What matters for the present work is this: the finite overlap operator $\mathcal{O}$ is not merely a computational approximation to an idealised convolution. It is, within the FSM frame, the *primary* object. The classical integral is its admissible endogenous extension. The relationship between them is not one of approximation but of **unfolding**: the classical form compresses a process; the finite operator makes that process explicit.

Where This Leads

For the reader who wishes to remain within the classical basin, the finite overlap operator stands as a useful generalisation. For the reader willing to cross the boundary, it becomes a gateway — an example of how a finite, process-based mathematics might be constructed from the ground up, starting from acts of distinction rather than idealised infinities. We leave both paths open.