AAC Physics: Structure-Preserving Framework for Exact, Reversible Discrete-Time Convolutions

Proprietary IP — Rich C. Young (Guru#1)

https://www.youtube.com/@GuruOne

Abstract

Memory-dependent discrete-time systems arise across physics, engineering, and machine learning—from viscoelasticity and control to differentiable programming. Standard recursive approaches accelerate convolution computation but suffer from drift, unstable backward passes, and numerical errors.

AAC Physics introduces a *structure-preserving, algebraically exact* framework for discrete convolutions with exponential-like memory kernels. The method achieves:

- Exact equivalence to full discrete convolution (no drift),
- $O(N)$ computational scaling with modest state storage,
- Algebraic reversibility, enabling machine-precision recovery of input sequences,
- Broad applicability across physics simulations, control systems, Neural ODEs, and gradient-based optimization.

1 Motivation and Background

Discrete convolutions with memory kernels appear in:

- **Viscoelasticity**: Stress–strain histories modeled via Prony-series exponentials.
- **Control systems**: Feedback with non-Markovian dynamics.
- **Neural ODEs**: Backpropagation through time with history-dependent states.

Challenges with existing methods:

- Naive summation is $O(N^2)$ and infeasible for long sequences.
- Recursive approximations introduce drift and reduce forward/backward accuracy.
- Standard recursive methods are often non-invertible, limiting exact adjoint calculations.

2 Core Concept: AAC Physics Framework

AAC Physics uses **abstract, IP-protected functions** to preserve structure and reversibility:

- **Forward update**
$$\text{state}_{n+1} = F(\text{state}_n, \text{input}_n)$$
- **Output computation**
$$\text{output}_n = G(\text{state}_n, \text{input}_n)$$
- **Backward recovery**
$$\text{input}_n = H(\text{state}_{n+1}, \text{state}_n)$$
- **State step-back**
$$\text{state}_n = I(\text{state}_{n+1}, \text{input}_n)$$

Key properties:

1. Machine-precision equivalence to full discrete convolution.
2. Linear $O(N)$ scaling for computation and memory.
3. Algebraic reversibility for exact backward reconstruction.
4. Proprietary coefficients protected within abstract functions F, G, H, I.

3 Performance and Accuracy

Table 1: Benchmark summary (IP-protected implementation).

Metric	AAC Physics	Standard Recursive	Naive Convolution
Forward Error	$\sim 10^{-16}$	$O(\Delta t)$ drift	Exact
Backward Error	$\sim 10^{-14}$ (quad)	Divergent	Exact
Complexity	$O(N)$	$O(N)$	$O(N^2)$
Speedup	$\sim 300\times$ vs naive	—	—

4 Multi-Mode Exponential Kernel Architecture

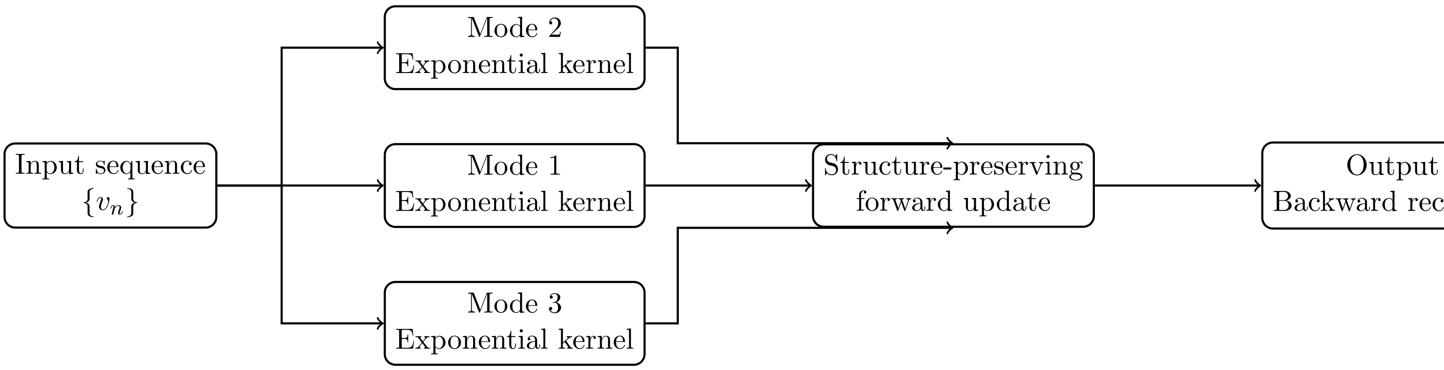

Figure 1: Multi-mode exponential kernel processing. Each kernel mode is updated independently and combined through a structure-preserving forward operator.

5 Forward and Backward Reversible Workflow

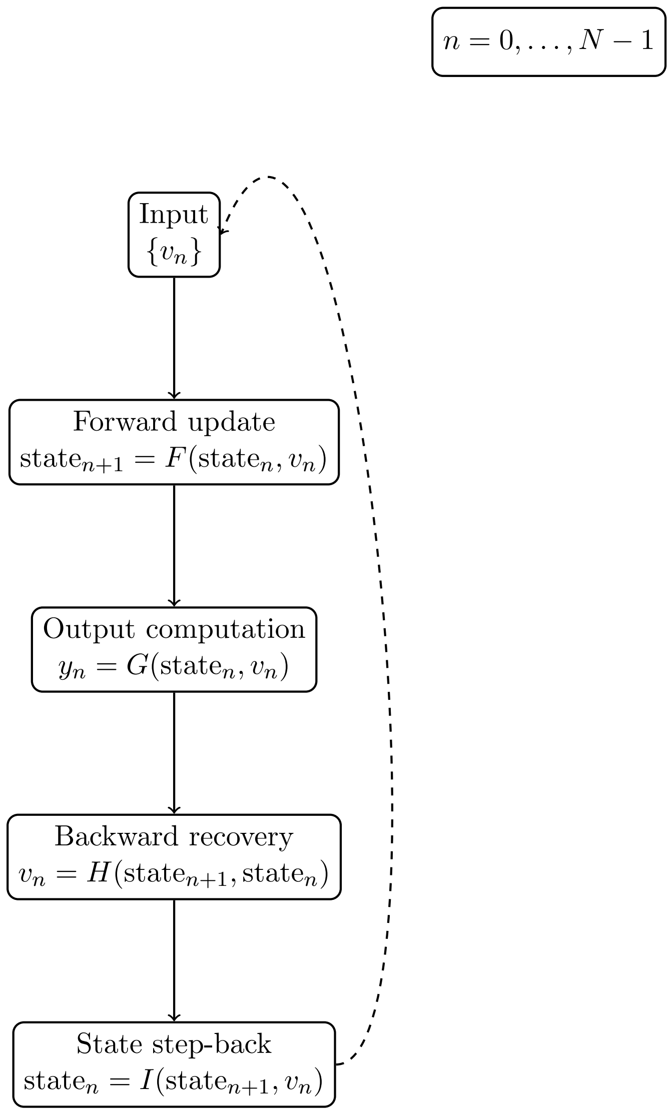

Figure 2: Structure-preserving forward and backward workflow. Algebraic reversibility enables exact recovery of input samples without storing full state histories.

6 Abstract Pseudocode

```
Forward pass:
for n = 0 .. N-1:
    state[n+1] = F(state[n], input[n])
    output[n]  = G(state[n], input[n])
```

```
Backward pass:
for n = N-1 .. 0:
    input[n] = H(state[n+1], state[n])
    state[n] = I(state[n+1], input[n])
```

7 Applications

- Physics simulations: viscoelasticity, fractional operators.
- Control and signal processing: non-Markovian feedback.
- Machine learning: Neural ODEs, adjoint computations.
- Industrial software: high-precision, memory-efficient simulations.

8 Summary

- $O(N)$ efficient, drift-free, machine-precision forward computation.
- Algebraic reversibility enables exact backward reconstruction.
- Broad applicability across physics, control, and machine learning.
- Fully IP-protected while transparently demonstrating performance and accuracy.