JANUS

Neuromorphic Architecture

Brain-Inspired Algorithmic Trading System
Mapping Neuroscience to Market Intelligence

Classification: Architecture Specification

Version: 1.0

Author: Jordan Smith
github.com/nuniesmith

Date: December 26, 2025

Neuromorphic Design Principles:

- **Cognitive Mapping:** Each brain region maps to a trading subsystem
- **Hierarchical Processing:** From sensory input to strategic planning
- **Parallel Computation:** Multiple regions process simultaneously
- **Homeostatic Regulation:** Self-balancing risk and reward
- **Fear-Conditioned Safety:** Emotional override for threat response

Abstract

JANUS implements a **neuromorphic architecture** that maps cognitive neuroscience principles to algorithmic trading. Each brain region's computational role is replicated in the system architecture, creating a biologically-inspired trading intelligence that combines:

- **Cortex**: Strategic planning and long-term memory (Manager Agent)
- **Hippocampus**: Episodic memory and experience replay (Worker Agent)
- **Basal Ganglia**: Action selection via Actor-Critic RL
- **Thalamus**: Attention gating and multimodal fusion
- **Prefrontal Cortex**: Logic, planning, and regulatory compliance
- **Amygdala**: Fear detection and emergency circuit breakers
- **Hypothalamus**: Homeostatic risk regulation
- **Cerebellum**: Motor control and optimal execution
- **Visual Cortex**: Pattern recognition via GAF and ViViT
- **Integration**: Brainstem coordination and lifecycle management

This architecture enables emergent intelligence through parallel processing, hierarchical control, and homeostatic self-regulation—principles proven effective in biological systems over millions of years of evolution.

Contents

1 Neuromorphic Design Philosophy

1.1 Why Brain-Inspired Architecture?

Traditional trading systems follow rigid, hierarchical designs. JANUS instead adopts principles from cognitive neuroscience:

1. **Parallel Processing**: Multiple brain regions process different aspects of market data simultaneously
2. **Hierarchical Abstraction**: Low-level pattern recognition feeds mid-level tactics which inform high-level strategy
3. **Homeostatic Regulation**: Self-balancing mechanisms maintain system health (like biological homeostasis)
4. **Emotional Override**: Fear systems can immediately halt trading when threats are detected
5. **Memory Consolidation**: Wake-sleep cycles transfer episodic experiences to long-term schemas
6. **Adaptive Learning**: Continuous learning at multiple timescales (fast hippocampal, slow cortical)

1.2 Neuroscience-to-Trading Mapping

Brain Region	Neuroscience Function	Trading Function
Cortex	Executive function, strategic planning, declarative memory	Manager agent, portfolio strategy, market knowledge
Hippocampus	Episodic memory, spatial navigation, memory replay	Worker agent, trade history, experience replay
Basal Ganglia	Action selection, habit formation, reward learning	Actor-Critic RL, action gating, Q-learning
Thalamus	Sensory relay, attention gating, arousal	Data fusion, attention mechanisms, signal filtering
Prefrontal	Logic, planning, impulse control, ethics	LTN constraints, compliance, goal decomposition
Amygdala	Fear conditioning, threat detection, emotional memory	Risk detection, circuit breakers, kill switch
Hypothalamus	Homeostasis, motivation, energy balance	Risk appetite, position sizing, cash management
Cerebellum	Motor coordination, procedural learning, error correction	Order execution, slippage prediction, PID control
Visual Cortex	Visual processing, feature extraction, object recognition	GAF encoding, ViViT, pattern recognition
Brainstem	Basic life functions, arousal/sleep cycles	System orchestration, wake/sleep coordination

Table 1: Neuroscience to Trading Mapping

2 Brain Region Architectures

2.1 Cortex: Strategic Planning & Long-term Memory

2.1.1 Neuroscience Background

The cerebral cortex handles executive function, strategic planning, and declarative (fact-based) memory. It operates on slow timescales, consolidating knowledge over days to years.

2.1.2 Trading Implementation

Directory: `src/janus/neuromorphic/cortex/`

Components:

- **Manager**: Feudal RL manager agent for high-level strategy
- **Memory**: Long-term knowledge consolidation and schema storage
- **Planning**: Scenario analysis, Monte Carlo, portfolio optimization

Key Responsibilities:

1. Set strategic goals (e.g., "maximize Sharpe ratio while maintaining drawdown <15%")
2. Generate subgoals for Worker agent (e.g., "accumulate position in AAPL over 2 hours")
3. Consolidate episodic memories into abstract schemas (e.g., "morning volatility regime")
4. Store declarative knowledge (e.g., "FOMC announcements increase volatility")

Mathematical Formulation:
Manager policy selects subgoals g for Worker:

$$g_t = \pi_{\text{Manager}}(s_t^{\text{high}}) \tag{1}$$

where s_t^{high} is high-level state (portfolio metrics, regime, time-to-horizon).
Value function:

$$V_{\text{Manager}}(s) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r_t^{\text{high}} \mid s_0 = s\right] \tag{2}$$

2.2 Hippocampus: Episodic Memory & Experience Replay

2.2.1 Neuroscience Background

The hippocampus rapidly encodes episodic memories and replays them during sleep at 10-20$\times$ speed. Pattern separation prevents interference. Sharp Wave Ripples (SWR) prioritize important experiences.

2.2.2 Trading Implementation

Directory: `src/janus/neuromorphic/hippocampus/`
Components:

- **Worker**: Feudal RL worker agent for tactical execution
- **Replay**: Prioritized Experience Replay (PER) buffer

- **Episodes**: Trade sequences and market events
- **SWR**: Sharp Wave Ripple simulator for compressed replay

Key Responsibilities:

1. Execute subgoals from Manager (e.g., "buy 100 shares incrementally")
2. Store trade experiences in episodic buffer
3. Prioritize replay based on TD-error + logic violations
4. Compress replay 10-20$\times$ during sleep (Backward service)

Mathematical Formulation:
Worker policy conditioned on subgoal g:

$$a_t = \pi_{\text{Worker}}(s_t^{\text{low}}, g_t) \tag{3}$$

Intrinsic reward for subgoal completion:

$$r_t^{\text{intrinsic}} = -||s_t - g_t||^2 \tag{4}$$

Prioritized replay sampling:

$$P(i) = \frac{p_i^\alpha}{\sum_j p_j^\alpha}, \quad p_i = |\delta_i| + \lambda_{\text{logic}} v_i + \lambda_{\text{reward}} |r_i| \tag{5}$$

2.3 Basal Ganglia: Action Selection & Reinforcement Learning

2.3.1 Neuroscience Background

The basal ganglia implement action selection via dual pathways: direct (Go) promotes actions, indirect (No-Go) inhibits them. This is the biological substrate for reinforcement learning.

2.3.2 Trading Implementation

Directory: `src/janus/neuromorphic/basal_ganglia/`
Components:

- **Actor**: Policy network for action distribution
- **Critic**: Value network for advantage estimation
- **Praxeological**: Go/No-Go signal computation

- **Selection**: Competitive action selection mechanisms

Key Responsibilities:

1. Generate action proposals (BUY, SELL, HOLD, sizes)
2. Compute action values (Q-values)
3. Gate actions through dual pathways (safety)
4. Maintain habit cache for frequent patterns

Mathematical Formulation:

Actor policy:

$$\pi_\theta(a|s) = \text{softmax}(\mathbf{W}_\pi \mathbf{h}(s) + \mathbf{b}_\pi) \tag{6}$$

Critic value estimate:

$$V_\omega(s) = \mathbf{W}_V \mathbf{h}(s) + b_V \tag{7}$$

Advantage:

$$A(s, a) = Q(s, a) - V(s) \tag{8}$$

Go signal (direct pathway):

$$\text{Go}(a) = \max\left(\mathbf{W}_{\text{direct}} \mathbf{h}(s)\right)_a \tag{9}$$

No-Go signal (indirect pathway):

$$\text{NoGo}(a) = \text{sigmoid}(\mathbf{W}_{\text{indirect}}[\mathbf{h}(s); \text{risk}; \text{VPIN}]) \tag{10}$$

Final action gate:

$$a_{\text{final}} = \begin{cases} a_{\text{proposed}} & \text{if NoGo}(a) < \tau_{\text{veto}} \\ \text{HOLD} & \text{otherwise} \end{cases} \tag{11}$$

2.4 Thalamus: Attention & Multimodal Fusion

2.4.1 Neuroscience Background

The thalamus acts as a sensory relay station, gating information flow to cortex based on attention and relevance. It integrates multimodal sensory inputs.

2.4.2 Trading Implementation

Directory: `src/janus/neuromorphic/thalamus/`

Components:

- **Attention**: Cross-attention mechanisms
- **Gating**: Sensory gating and relevance filtering
- **Routing**: Dynamic information routing
- **Fusion**: Price, volume, orderbook, sentiment fusion

Key Responsibilities:

1. Gate incoming market data (filter noise)
2. Fuse multiple data modalities (price, volume, text)
3. Route relevant information to appropriate regions
4. Implement attention mechanisms for saliency

Mathematical Formulation:
Gated cross-attention:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V} \tag{12}$$

Gating scalar:

$$\lambda_{\text{gate}} = \mathrm{sigmoid}(\mathbf{W}_g[\mathbf{h}_m; \mathbf{h}_n] + b_g) \tag{13}$$

Fused representation:

$$\mathbf{h}_{\text{fused}} = \mathbf{h}_m + \lambda_{\text{gate}} \cdot \text{Attention}(\mathbf{h}_m, \mathbf{h}_n, \mathbf{h}_n) \tag{14}$$

2.5 Prefrontal Cortex: Logic, Planning & Compliance

2.5.1 Neuroscience Background

The prefrontal cortex implements logical reasoning, impulse control, and ethical decision-making. It’s the "executive" that enforces rules and long-term goals.

2.5.2 Trading Implementation

Directory: `src/janus/neuromorphic/prefrontal/`
Components:

- **LTN**: Logic Tensor Networks for rule encoding
- **Conscience**: Compliance constraints (wash sale, risk limits)
- **Planning**: Goal decomposition and plan synthesis

- **Goals**: Goal hierarchy management

Key Responsibilities:

1. Encode trading rules as differentiable logic
2. Enforce regulatory compliance (wash sale, position limits)
3. Block actions violating constraints
4. Decompose high-level goals into actionable plans

Mathematical Formulation:
LTN predicate grounding:

$$\mathcal{G}(P)(\mathbf{x}) = \mathrm{sigmoid}(\mathbf{W}_P\mathbf{x} + b_P) \in [0, 1] \tag{15}$$

Łukasiewicz conjunction:

$$\mathcal{G}(A \wedge B) = \max(0, \mathcal{G}(A) + \mathcal{G}(B) - 1) \tag{16}$$

Wash sale constraint:

$$\forall t, k \in [1, 30] : \neg(\text{SaleAtLoss}(t) \wedge \text{Buy}(t + k)) \tag{17}$$

Satisfiability:

$$\text{SatAgg}(\mathcal{K}) = \left(\frac{1}{m}\sum_{i=1}^{m} \mathcal{G}(\phi_i)^p\right)^{1/p} \tag{18}$$

2.6 Amygdala: Fear, Threat Detection & Circuit Breakers

2.6.1 Neuroscience Background

The amygdala detects threats and triggers immediate fear responses, overriding rational planning when danger is present. Fear-conditioned memories persist long-term.

2.6.2 Trading Implementation

Directory: `src/janus/neuromorphic/amygdala/`
Components:

- **Fear**: Fear-conditioned inhibition network (FNI-RL)
- **VPIN**: Volume-synchronized toxicity detection
- **Circuit Breakers**: Kill switch, position freeze, cancel all

- **Threat Detection**: Anomaly, flash crash, black swan detection

Key Responsibilities:

1. Detect market panic and flash crashes
2. Trigger emergency circuit breakers
3. Override all other systems in extreme conditions
4. Learn fear-conditioned responses to past disasters

Mathematical Formulation:
VPIN (Volume-Synchronized Probability of Informed Trading):

$$\text{VPIN}_t = \frac{\sum_{i=1}^{n} |V_{\text{buy},i} - V_{\text{sell},i}|}{\sum_{i=1}^{n} V_i} \tag{19}$$

Fear activation:

$$f_{\text{fear}}(s) = \text{sigmoid}(\mathbf{W}_f[\text{VPIN}; \sigma_{\text{vol}}; \Delta p_{\text{max}}] + b_f) \tag{20}$$

Circuit breaker trigger:

$$\text{KillSwitch} = \begin{cases} \text{ACTIVATE} & \text{if } f_{\text{fear}} > \tau_{\text{fear}} \text{ OR VPIN} > \tau_{\text{VPIN}} \\ \text{STANDBY} & \text{otherwise} \end{cases} \tag{21}$$

2.7 Hypothalamus: Homeostasis & Risk Appetite

2.7.1 Neuroscience Background

The hypothalamus maintains homeostasis—internal balance of temperature, hunger, thirst, etc. It regulates motivation and energy expenditure.

2.7.2 Trading Implementation

Directory: `src/janus/neuromorphic/hypothalamus/`

Components:

- **Homeostasis**: Balance tracking and deviation correction
- **Position Sizing**: Kelly criterion, volatility scaling
- **Risk Appetite**: Dynamic risk tolerance adaptation
- **Energy**: Capital allocation and cash reserves

Key Responsibilities:

1. Maintain target portfolio balance (setpoints)
2. Adjust position sizes based on volatility and drawdown
3. Regulate risk appetite (fear vs. greed)
4. Ensure cash reserves and leverage limits

Mathematical Formulation:
Kelly criterion (fractional):

$$f^* = \frac{p(b+1)-1}{b}, \quad \text{position size} = \frac{f^*}{2} \cdot \text{capital} \tag{22}$$

Volatility scaling:

$$\text{size}_{\text{adjusted}} = \text{size}_{\text{base}} \cdot \frac{\sigma_{\text{target}}}{\sigma_{\text{current}}} \tag{23}$$

Drawdown scaling:

$$\text{size}_{\text{DD}} = \text{size}_{\text{base}} \cdot \max\left(0.1, 1 - \frac{\text{DD}_{\text{current}}}{\text{DD}_{\text{max}}}\right) \tag{24}$$

Homeostatic correction:

$$\Delta\text{allocation} = K_p \cdot (\text{target} - \text{current}) + K_d \cdot \frac{d(\text{target} - \text{current})}{dt} \tag{25}$$

2.8 Cerebellum: Motor Control & Execution

2.8.1 Neuroscience Background

The cerebellum coordinates fine motor control, learns procedural skills, and predicts sensory consequences of actions (forward models).

2.8.2 Trading Implementation

Directory: `src/janus/neuromorphic/cerebellum/`
Components:

- **Execution**: Order routing, TWAP/VWAP algorithms
- **Impact**: Almgren-Chriss optimal execution
- **Forward Models**: Latency compensation, fill prediction
- **Error Correction**: PID control, adaptive feedback

Key Responsibilities:

1. Route orders to exchanges with minimal slippage
2. Predict and minimize market impact
3. Compensate for execution latency (Smith predictor)
4. Learn from execution errors and adapt

Mathematical Formulation:
Almgren-Chriss optimal trajectory:

$$x_t = X \cdot \frac{\sinh(\kappa(T-t))}{\sinh(\kappa T)}, \quad \kappa = \sqrt{\frac{\eta\sigma}{\tau}} \tag{26}$$

Market impact:

$$\text{Impact} = \eta \cdot \sigma \cdot \sqrt{\frac{v}{V_{\text{avg}}}} \tag{27}$$

Smith predictor (latency compensation):

$$u(t) = K_c \left[e(t) + \frac{1}{\tau_I} \int e(\tau) d\tau + \tau_D \frac{de(t)}{dt} \right] + \hat{p}(t + \Delta t) \tag{28}$$

Execution error:

$$\epsilon_{\text{exec}} = |p_{\text{actual}} - p_{\text{predicted}}| \tag{29}$$

2.9 Visual Cortex: Pattern Recognition & Vision

2.9.1 Neuroscience Background

The visual cortex processes images hierarchically: V1 detects edges, V2 detects shapes, V4 detects objects. It implements hierarchical feature extraction.

2.9.2 Trading Implementation

Directory: `src/janus/neuromorphic/visual_cortex/`

Components:

- **Eyes**: Data ingestion, preprocessing, streaming
- **GAF**: Gramian Angular Fields (GASF, GADF, DiffGAF)
- **ViViT**: Video Vision Transformer for spatiotemporal patterns
- **Visualization**: UMAP, GradCAM, saliency maps

Key Responsibilities:

1. Ingest and preprocess raw market data
2. Transform time series to visual manifolds (GAF)
3. Extract spatiotemporal patterns (ViViT)
4. Visualize learned representations (UMAP)

Mathematical Formulation:
GAF normalization:

$$\tilde{x}_i = \tanh\left(\frac{x_i - \min(X)}{\max(X) - \min(X) + \epsilon} \cdot \alpha + \beta\right) \tag{30}$$

GASF:

$$\text{GASF}_{i,j} = \cos(\phi_i + \phi_j), \quad \phi_i = \arccos(\tilde{x}_i) \tag{31}$$

GADF:

$$\text{GADF}_{i,j} = \sin(\phi_i - \phi_j) \tag{32}$$

ViViT patch embedding:

$$\mathbf{z}^{(0)}_{f,i,j} = \mathbf{E} \cdot \text{flatten}(\mathcal{V}_{f,i:i+P,j:j+P}) + \mathbf{p}_{f,i,j} \tag{33}$$

2.10 Integration: Brainstem & Global Coordination

2.10.1 Neuroscience Background

The brainstem controls basic life functions, arousal/sleep cycles, and global state coordination. It's the "operating system" of the brain.

2.10.2 Trading Implementation

Directory: `src/janus/neuromorphic/integration/`
Components:

- **Workflow**: State machine orchestration
- **State**: Global state management, message bus
- **API**: REST, gRPC, WebSocket interfaces
- **Engine**: Wake-sleep cycle coordination

Key Responsibilities:

1. Coordinate wake (Forward) and sleep (Backward) cycles

2. Manage global system state
3. Route messages between brain regions
4. Expose external APIs

3 Information Flow Diagrams

3.1 Wake State (Forward Service)

```
Market Data → Visual Cortex (GAF/ViViT) → Thalamus (Fusion)
                                               ↓
                                          Cortex (Manager)
                                               ↓
                                          Hippocampus (Worker)
                                               ↓
                                          Basal Ganglia (Actor-Critic)
                                               ↓
                                          Prefrontal (LTN Check)
                                               ↓
                                          Amygdala (Fear Check) ⟶ Circuit Breaker?
                                               ↓
                                          Hypothalamus (Size Adjust)
                                               ↓
                                          Cerebellum (Execute Order)
                                               ↓
                                          Exchange
```

3.2 Sleep State (Backward Service)

```
Hippocampus (Episodic Buffer) → SWR Replay (10-20x speed)
                                     ↓
                                SWR Replay (10-20x speed)
```

4 Implementation Guide

4.1 Directory Structure

```
src/janus/neuromorphic/
├── lib.rs                    # Main library entry point
├── common/                   # Shared types and utilities
├── cortex/                   # Strategic planning & LTM
│   ├── manager/             # Feudal RL manager
│   ├── memory/              # Consolidation, schemas
│   └── planning/            # Scenario analysis
├── hippocampus/             # Episodic memory & replay
│   ├── worker/              # Feudal RL worker
│   ├── replay/              # PER buffer
│   ├── episodes/            # Trade sequences
│   └── swr/                 # Sharp wave ripples
├── basal_ganglia/           # Action selection & RL
│   ├── actor/               # Policy network
│   ├── critic/              # Value network
│   ├── praxeological/       # Go/No-Go signals
│   └── selection/           # Action selection
├── thalamus/                # Attention & fusion
│   ├── attention/           # Cross-attention
│   ├── gating/              # Sensory gates
│   ├── routing/             # Information routing
│   └── fusion/              # Multimodal fusion
├── prefrontal/              # Logic & compliance
│   ├── ltn/                 # Logic Tensor Networks
│   ├── conscience/          # Compliance rules
│   ├── planning/            # Goal decomposition
│   └── goals/               # Goal management
├── amygdala/                # Fear & circuit breakers
│   ├── fear/                # FNI-RL network
│   ├── vpin/                # Toxicity detection
│   ├── circuit_breakers/    # Kill switch
│   └── threat_detection/    # Anomaly detection
├── hypothalamus/            # Homeostasis & risk
│   ├── homeostasis/         # Balance tracking
│   ├── position_sizing/     # Kelly, vol scaling
```

```
│   ├── risk_appetite/      # Dynamic tolerance
│   └── energy/             # Capital allocation
├── cerebellum/             # Motor control & execution
│   ├── execution/          # Order routing
│   ├── impact/             # Almgren-Chriss
│   ├── forward_models/     # Latency compensation
│   └── error_correction/   # PID control
├── visual_cortex/          # Pattern recognition
│   ├── eyes/               # Data ingestion
│   ├── gaf/                # GAF transformation
│   ├── vivit/              # ViViT model
│   └── visualization/      # UMAP, GradCAM
└── integration/            # System coordination
    ├── workflow/           # State machines
    ├── state/              # Global state
    ├── api/                # External APIs
    └── engine/             # Orchestration
```

4.2 Implementation Checklist

1. **Phase 1: Core Infrastructure (Weeks 1-2)**
   - ☐ Set up neuromorphic module structure
   - ☐ Implement common types and error handling
   - ☐ Create inter-region message bus
   - ☐ Set up integration/engine orchestrator
2. **Phase 2: Visual Processing (Weeks 3-4)**
   - ☐ Implement Visual Cortex data ingestion
   - ☐ Implement GAF transformation (GASF, GADF)
   - ☐ Integrate ViViT model (ONNX or tch-rs)
   - ☐ Add UMAP visualization
3. **Phase 3: Decision Making (Weeks 5-7)**
   - ☐ Implement Basal Ganglia Actor-Critic
   - ☐ Implement Prefrontal LTN constraints
   - ☐ Implement Thalamus fusion mechanisms

- ☐ Connect visual → decision pipeline

4. **Phase 4: Memory Systems (Weeks 8-10)**
   - ☐ Implement Hippocampus episodic buffer
   - ☐ Implement Prioritized Experience Replay
   - ☐ Implement Cortex schema consolidation
   - ☐ Implement SWR compressed replay
5. **Phase 5: Safety & Control (Weeks 11-12)**
   - ☐ Implement Amygdala fear network
   - ☐ Implement circuit breakers and kill switch
   - ☐ Implement Hypothalamus homeostasis
   - ☐ Implement Cerebellum execution control
6. **Phase 6: Integration & Testing (Weeks 13-14)**
   - ☐ Connect all brain regions
   - ☐ Implement wake-sleep cycle coordination
   - ☐ End-to-end integration tests
   - ☐ Performance optimization

5 Architectural Invariants

5.1 Safety-Critical Invariants

1. **Amygdala Override**: Fear system ALWAYS overrides all other regions
2. **Prefrontal Veto**: LTN constraints MUST block non-compliant actions
3. **No Panic**: No `panic!()`, `unwrap()`, or `expect()` in production code
4. **Fail-Safe**: Circuit breakers must be fail-safe (default to HALT)
5. **Homeostasis**: Cash reserves must never fall below 20%

5.2 Performance Invariants

1. **Forward Latency**: Visual Cortex $\rightarrow$ Decision <100ms
2. **Backward Throughput**: Process >10k experiences per sleep cycle
3. **Memory Efficiency**: Hippocampal buffer <10k transitions (FIFO eviction)
4. **Parallel Processing**: Brain regions process concurrently

5.3 Learning Invariants

1. **Dual Timescale**: Fast hippocampal learning, slow cortical consolidation
2. **Recall Gating**: Cortical updates gated by recall strength AND logic validity
3. **Priority Replay**: Replay prioritized by TD-error + logic violations + reward
4. **Schema Formation**: Clusters detected via UMAP + DBSCAN

References

[1] Jordan Smith, "JANUS Forward: Wake State Logic Trading Algorithm," 2025.

[2] Jordan Smith, "JANUS Backward: Sleep State Memory Management," 2025.

[3] Dayan, Hinton, "Feudal Reinforcement Learning," NIPS 1992.

[4] Sutton, Barto, "Reinforcement Learning: An Introduction," 2nd Ed., 2018.

[5] Badreddine et al., "Logic Tensor Networks," Artificial Intelligence, 2022.

[6] Schaul et al., "Prioritized Experience Replay," ICLR 2016.

[7] Foster, Wilson, "Reverse Replay of Behavioural Sequences in Hippocampal Place Cells," Nature 2006.

[8] "Fear-Conditioned Inhibition in Reinforcement Learning," 2020.

[9] Sterling, Eyer, "Allostasis: A New Paradigm to Explain Arousal Pathology," 1988.

[10] Almgren, Chriss, "Optimal Execution of Portfolio Transactions," Journal of Risk, 2000.

[11] Easley et al., "Flow Toxicity and Liquidity in a High-frequency World," Review of Financial Studies, 2012.

[12] Wang, Oates, "Encoding Time Series as Images," AAAI 2015.

[13] Arnab et al., "ViViT: A Video Vision Transformer," ICCV 2021.

[14] McInnes et al., "UMAP: Uniform Manifold Approximation and Projection," 2018.

[15] Kandel et al., "Principles of Neural Science," 6th Ed., 2021.