HVTS-LIQUIDITY PROTOCOL

NeuralNetwork-GMMA Multi-Asset Edition

High Velocity Trading as a Service

Product Codename	HVTS-Liquidity Protocol — NeuralNetwork-GMMA Multi-Asset Edition
Asset Class	Cryptocurrency — Multi-Asset Spot (BTC, ETH, SOL, XRP, ADA, DOGE, POL, DOT, AVAX, BNB + more)
Platform	Gate.io Spot Market — Python CCXT Async API
Architecture	Async Multi-Strategy — 7 Concurrent Engines + Triple OR Signal Protocol
Codebase	19,994+ lines across 19 production modules
Signal Engine	Triple OR Protocol — Neural Network (per-symbol) · GMMA H1/H4/Daily
NN Architecture	BiLSTM + Multi-Head Attention + Residual Dense — 4-output simultaneous prediction
New Modules	NeuralSignalPerSymbol · PivotZoneAnalysis · MultiUserConfig · GateioExchange
Multi-User	Full per-user isolation — exchange credentials, symbols, strategies, zones
Classification	Proprietary Algorithmic Trading Software
System Version	v7.0 (15-Strategy Triple Consensus with 9-Stage Deep Scanner & Global Trend Reset)
Document Date	June 2026

1 Executive Summary

19,994+	7	11+	4	$75K
Lines of Code 19 production modules	Active Strategies Concurrent & independent	Supported Assets BTC · ETH · SOL + more	Neural Outputs Trend · Price Volatility · Conf.	Fair Market Value Source code sale

HVTS-Liquidity Protocol (NeuralNetwork-GMMA Multi-Asset Edition) is a sophisticated, production-grade algorithmic trading system purpose-built for high-frequency multi-asset spot trading on Gate.io. Engineered exclusively for cryptocurrency spot markets, this system brings institutional-grade infrastructure to the Gate.io ecosystem through its native CCXT async integration and per-symbol neural signal architecture.

Unlike its Bitcoin-only predecessor, HVTS-Liquidity Protocol operates across eleven or more cryptocurrency pairs simultaneously, each governed by its own independent neural signal sourced from a dedicated per-symbol SQLite database. The system's Triple OR Signal Protocol — combining per-symbol Neural Network predictions with GMMA Daily, H4, and H1 overlays — provides asymmetric entry precision while the ATH-based 5-Unit Zone Fund Allocation System enforces disciplined capital deployment across five market discount zones.

Feature	Value
Trading Assets	Multi-asset spot: BTC, ETH, SOL, XRP, ADA, DOGE, POL, DOT, AVAX, BNB, XPL (11+ pairs)
Timeframe	Configurable — 1m to 1d; 4-hour OHLCV for neural; ATH on daily bars (365-candle lookback)
Active Strategies	7 concurrent: DCA, Grid, AMM, Signal, Scalper, Pivot, Accumulator
Signal Engine	Triple OR Protocol: Neural Network (per-symbol) + GMMA H1/H4/Daily
Neural Architecture	BiLSTM + Multi-Head Attention (4 heads) + Residual Dense — Keras Functional API
Neural Outputs	Trend classification (5-class) · Price forecast (5-step) · Volatility · Confidence
Entry Rule	ANY ONE of: (Neural + GMMA Daily) OR (GMMA H4 + GMMA Daily) OR (GMMA H1 + GMMA Daily)
Emergency Exit	GMMA Daily turns bearish → close positions at minimal profit (even 0%)
Force Close	ALL FOUR signals bearish → close ALL positions + cancel ALL pending orders
Zone System	5-Unit ATH-based allocation: 5 zones × 20% each — all zones tradeable
Multi-User	Full per-user isolation: exchange credentials, symbols, strategies, zones
License System	Node-locked with optional expiry — LicenseManager + activation workflow
Replacement Cost	$60,000 – $120,000 (rebuild from scratch)
Private Sale Value	$35,000 – $75,000 (source code + IP transfer)
SaaS Potential	$200 – $600 per subscriber per month
Prop Firm Licence	$50,000 – $150,000 + royalties

2 Introduction & Background

The Problem

Retail and semi-professional traders attempting to automate multi-asset cryptocurrency spot trading face a compounded challenge: building a reliable, concurrent system that can manage dozens of symbols independently — each with its own entry logic, position sizing, and risk controls — requires deep expertise in quantitative finance, asynchronous systems programming, exchange API integration, and machine-learning signal consumption. Off-the-shelf solutions are typically single-asset, single-strategy, or lack the infrastructure required for continuous unattended operation across a diversified portfolio.

The Solution — NeuralNetwork-GMMA Multi-Asset Edition

HVTS-Liquidity Protocol addresses this gap with a layered, per-symbol multi-signal architecture built natively for Gate.io's spot market. The system introduces several architectural advances:

- Per-Symbol Neural Network — each trading pair reads from its own dedicated SQLite database, enabling fully independent ML-driven signals per asset.
- Triple OR Signal Protocol — an asymmetric consensus gate: any one of three signal-pair combinations unlocks entry; the GMMA Daily alone governs emergency exit.
- GMMA Multi-Timeframe Overlay — Guppy Multiple Moving Average computed on H1, H4, and Daily bars, providing trend confirmation across three timeframes simultaneously.
- 5-Unit ATH-Based Zone System — all-in capital strategy divided into five 20% pools mapped to ATH discount zones.
- 7 Concurrent Strategies — DCA, Grid, AMM, Signal, Scalper, Pivot, and Accumulator all share the same signal bus but operate with independent position tracking.
- Full Multi-User Architecture — per-user exchange credentials, symbol lists, strategy toggles, zone overrides, and AMM configurations.
- Gate.io Native Design — built exclusively for Gate.io spot using CCXT's async Gate connector.

3 Solution Overview — 19-Module Architecture

HVTS-Liquidity Protocol is a Python-based autonomous trading system connecting to Gate.io via the CCXT async library. Upon launch it validates the license, initialises all databases, loads per-user configurations, validates symbols against live Gate.io markets, then runs seven trading strategies per symbol across all active users as fully concurrent asyncio tasks.

Module	Lines	Category	Primary Responsibility
trader.py	1,488	Orchestration	Per-user trading loop: symbol dispatch, signal coordination, position lifecycle
strategies.py	1,457	Strategy Engine	BaseStrategy + 7 strategy classes: DCA, Grid, AMM, Signal, Scalper, Pivot, Accumulator
signal_consensus_utils.py	1,695	Signal Engine	Triple OR Protocol; GMMA direction resolver; SuperTrend bridge; consensus cache
neural_signal.py	1,518	Signal Engine	Per-symbol neural provider: reads dedicated DB per asset, binary state machine
neural_signal_per_symbol.py	1,073	Signal Engine	Per-symbol provider factory, SYMBOL_NEURAL_DB_MAP, persistent state
SuperTrendOracleIndicator_M5.py	2,028	Signal Engine	Fibonacci oracle resampling; SignalType classification; OracleManager
main.py	1,164	Orchestration	System startup, license check, triple signal init, watchdog, user trader loops
config.py	1,144	Config/Infra	ConfigurationManager: all strategy params, neural config, exchange config
user_default_config.py	813	Config/Infra	Hard-coded defaults: per-symbol neural DB map, zone config, GMMA/SuperTrend params
multi_user_config.py	833	Config/Infra	MultiUserConfigurationManager: per-user exchange/symbol/strategy isolation
telegram_manager.py	1,329	Reporting	Real-time Telegram: trade events, signal changes, P&L; summaries, health alerts
reporting.py	726	Reporting	Enhanced reporting: P&L; summaries, hourly reports, zone transition alerts
state_manager.py	793	Infrastructure	SQLite-backed persistence: positions, portfolio, neural state, backups
pivot_analysis.py	288	Infrastructure	ATH-based zone analysis: 5-Unit fund allocation, zone boundary calculations
gateio_exchange.py	974	Infrastructure	Gate.io CCXT async connector: order management, balance fetch, market validation
db_exchange.py	933	Infrastructure	ExchangeManager: multi-exchange abstraction layer, OHLCV fetch, price streaming
license_manager.py	847	Infrastructure	Node-locked license validation, machine ID generation, expiry, activation workflow
user_manager.py	373	Infrastructure	UserManager: active user loading, multi-user database operations
run.py	518	Entry Point	CLI entry point: argument parsing, license pre-check, neural DB check, quick-start
TOTAL — 19 modules	19,994+		Est. 700–1,100+ hours specialist development

4 Seven Independent Trading Strategies

Each strategy operates as a fully independent asyncio task per symbol per user, with its own position tracking, profit targets, and Triple OR signal gate. All seven share the same Oracle signal bus while maintaining separate database tables and execution contexts.

01 DCA Strategy

Dollar-cost averaging cycles triggered on confirmed neural or GMMA direction changes. Averaging dips are constrained by the zone pool allocation (20% of available balance).

- Triple OR entry: any one of (Neural+Daily) OR (H4+Daily) OR (H1+Daily) bullish
- Emergency exit: GMMA Daily bearish — close at minimal profit
- Force-close: ALL FOUR signals bearish → cancel all limit orders + close all positions
- Zone-aware position sizing via 5-Unit ATH allocation system

02 Grid Strategy

Fixed-interval grid placing staggered buy and sell limit orders above and below current price. Grid spacing is configurable (fixed percentage or ATR-based).

- Triple OR consensus governs grid activation and reactivation
- ATR-based dynamic grid spacing adapts to current volatility
- Auto-regrid on price deviation beyond configurable threshold

03 AMM Strategy

Automated market-maker style range-bound grid that concentrates liquidity within a configured price band (±5% default).

- Dynamic band adjustment on price deviation beyond AMM_ADJUSTMENT_THRESHOLD
- Per-user AMM configuration: USDT per grid, number of grids, TP percent
- Triple OR gate prevents AMM deployment during bearish consensus

04 Signal Strategy

Pure signal-following strategy that opens positions exclusively on strong bullish signal states.

- Entry only on strong neural + GMMA daily bullish alignment
- Highest profit target of all 7 strategies (6% default)
- Rapid exit on any signal deterioration

05 Scalper Strategy

Short-duration scalping strategy targeting rapid 3% profit captures within tight price ranges.

- 3% default profit target — lowest of all strategies for fastest turnover
- ATR-based entry/exit range adapts to intraday volatility
- Immediate market-order exit on GMMA Daily bearish

06 Pivot Strategy

ATH-based pivot zone entry strategy that deploys capital exclusively at zone transition boundaries.

- Zone-specific entries at ATH discount levels: 25%, 50%, 75%, 90%
- Momentum requirement per zone — e.g., Extreme Discount requires 60% confidence
- 4% default profit target; closes on zone exit or GMMA Daily bearish

07 Accumulator Strategy

Multi-level DCA accumulator that staggers entries across configurable dip levels as price moves against the initial position.

- Configurable dip levels (e.g., -2%, -5%, -8% from entry) for staged accumulation
- Per-position cooldown prevents over-averaging during volatile drops
- 5% default profit target; emergency exit on GMMA Daily bearish with no profit floor

5 HVTS-NeuralNetwork — Deep Learning Signal Engine

HVTS-NeuralNetwork is a production-grade, autonomous cryptocurrency market analysis system built on a dual-branch deep learning architecture. The system ingests four-hour OHLCV candle data for ten major cryptocurrency pairs from Gate.io, engineers a 24-feature technical indicator space, and trains a multi-output neural network that simultaneously predicts market trend classification, multi-step price forecasts, volatility estimates, and confidence scores.

5.1 Neural Network Architecture

Branch A — BiLSTM (128 units)	Processes sequence forward & backward; produces 256-dim representation per timestep
Multi-Head Self-Attention (4 heads)	Learns which of 75 timesteps are most relevant. Residual Add & Norm wraps attention
BiLSTM (64 units)	Collapses attended sequence into single 128-dim context vector
Branch B — Dense + Residual	1,800-dimensional vector; Swish activation; 256→256→256→128→64 units with skip connections
Branch Fusion	128+64 = 192-dim merged; two Dense layers → four independent output heads
Output: Trend	Dense(5) Softmax — 5-class: STRONG_BEAR → STRONG_BULL
Output: Price	Dense(5) Linear — % price change at steps 1–5 ahead
Output: Volatility	Dense(1) Linear — Expected mean absolute return
Output: Confidence	Dense(2) Softmax — Binary: high / low confidence

5.2 Feature Engineering — 24-Feature Technical Indicator Space

Price Dynamics	returns, log_returns, price_range
Momentum	rsi_14, rsi_28, macd, macd_signal, macd_histogram
Moving Averages	ma_5/10/20/50/100 + price_vs_ma deviations (5 pairs)
Volatility	volatility_20, volatility_50, ATR(14)
Volume	volume_ma_20, volume_ratio
Regime	trend_strength (polyfit slope over 20 closes)
Scaler	RobustScaler (median + IQR) — resistant to crypto outliers

5.3 Training & Regularization

- Temporal split: 75% train / 15% validation / 10% test (chronological, no shuffle)
- Dropout 0.2–0.5, Recurrent Dropout 0.2–0.3, L1=1e-6, L2=5e-5 weight decay
- Batch Normalization on all Dense layers; Early Stopping patience=20; Gradient Clipping clipnorm=1.0
- Retraining every 12 hours; signal reports every 4 hours; circuit breaker after 3 failures (6h timeout)
- Synthetic GBM fallback ensures signal continuity during exchange downtime

6 Signal Engine & Architecture

Triple OR Signal Protocol

HVTS-Liquidity Protocol replaces the original dual-signal gate with the Triple OR Signal Protocol — a four-source asymmetric consensus framework. Three signal-pair combinations each independently authorise entry, while a single condition (GMMA Daily bearish) triggers emergency exit. This asymmetry maximises entry opportunity while maintaining strict downside discipline.

Signal Source	Module	Timeframe	Description
Neural Network (per-symbol)	neural_signal.py / neural_signal_per_symbol.py	External DB	BiLSTM-Attention model; 5-class trend; per-symbol SQLite DB
GMMA Daily	signal_consensus_utils.py / gmma_manager.py	D1	Guppy Multiple Moving Average on daily bars; primary exit gate
GMMA H4	signal_consensus_utils.py / gmma_manager.py	H4	GMMA computed on 4-hour bars; pairs with Daily for second OR
GMMA H1	signal_consensus_utils.py / gmma_manager.py	H1	GMMA computed on 1-hour bars; pairs with Daily for third OR
SuperTrend Oracle (optional)	SuperTrendOracleIndicator_M5.py	15m/1h	Fibonacci oracle with polynomial regression; optional overlay

Condition	Rule	Outcome
ENTRY	(Neural + GMMA Daily) OR (GMMA H4 + GMMA Daily) OR (GMMA H1 + GMMA Daily) bullish	Any one pair sufficient
EMERGENCY EXIT	GMMA Daily turns bearish	Close positions at minimal profit (even 0%)
FORCE CLOSE	ALL FOUR signals bearish	Close ALL positions; cancel ALL pending orders
NEUTRAL	Signal present but no consensus	No action — hold existing positions

7 Infrastructure & Reliability

DATABASE — SQLite + WAL Mode

Thread-safe concurrent access via DatabaseManager singleton. WAL journal mode allows multiple strategy tasks to read without blocking writes. 6-hour automated backups with checksum-validated state recovery.

STATE — Full State Persistence

StateManager serialises positions, portfolio, pending orders, and neural signal state to disk. neural_state_per_symbol table preserves the last-known binary BULLISH/BEARISH state across all restarts.

LICENSE — Node-Locked System

LicenseManager validates a machine-locked license.key on every startup. Machine ID derived from MAC address, hostname, and OS-specific identifiers. Hard abort and Telegram alert on validation failure. Optional expiry date.

ZONES — ATH-Based 5-Unit Zone System

PivotZoneAnalysis fetches 365-day daily OHLCV from Gate.io, computes ATH, and maps current price to one of five discount zones. Each zone receives exactly 20% of available balance. ATH cache expires hourly.

MULTI-USER — Per-User Isolation

MultiUserConfigurationManager loads independent configurations per user: separate exchange credentials, symbol lists, strategy toggles, zone overrides, AMM parameters, and per-symbol neural overrides.

EXCHANGE — Gate.io Spot Native

GateioExchange uses ccxt.gate() with standard slash-format symbols (BTC/USDT). Rate limiting (1,000ms) and timeout (60s) configurable. Symbols validated against live Gate.io spot markets on startup.

TELEGRAM — Real-Time Notifications

Full event coverage: startup, shutdown, trade executions, signal changes, zone transitions, emergency exits, force-close events, hourly P&L; summaries, health alerts. HTML-formatted with emoji indicators.

WATCHDOG — System Monitor

Asyncio watchdog monitors the main trading loop. If unresponsive for 600+ seconds, a warning is logged. Graceful shutdown on SIGTERM/SIGINT with final state save and Telegram notification.

8 Codebase Metrics — 19,994+ Lines of Production Python

Module	Lines	Category	Primary Responsibility
SuperTrendOracleIndicator_M5.py	2,028	Signal Engine	Fibonacci oracle; SignalType classification; OracleManager
signal_consensus_utils.py	1,695	Signal Engine	Triple OR Protocol; GMMA resolver; consensus cache; force-close logic
trader.py	1,488	Orchestration	Per-user loop: symbol dispatch, neural check, position lifecycle
neural_signal.py	1,518	Signal Engine	Per-symbol neural provider; binary state machine; DB map; confidence filter
strategies.py	1,457	Strategy Engine	BaseStrategy + 7 concrete strategy classes with full Triple OR integration
telegram_manager.py	1,329	Reporting	Real-time Telegram: HTML messages, emoji indicators, all trade/signal events
config.py	1,144	Config/Infra	ConfigurationManager: neural, exchange, zone, AMM, grid, dynamic sizing
neural_signal_per_symbol.py	1,073	Signal Engine	Per-symbol provider factory; SYMBOL_NEURAL_DB_MAP; per-symbol state
gateio_exchange.py	974	Infrastructure	Gate.io CCXT async connector; credential validation; order management
db_exchange.py	933	Infrastructure	ExchangeManager abstraction: multi-exchange, balance, price, orders
license_manager.py	847	Infrastructure	Node-locked validation; machine ID; expiry; Telegram abort on failure
multi_user_config.py	833	Config/Infra	MultiUserConfigurationManager: per-user exchange/symbol/strategy isolation
user_default_config.py	813	Config/Infra	Hard-coded defaults: symbol DB map, zone config, GMMA/SuperTrend params
state_manager.py	793	Infrastructure	SQLite persistence: positions, portfolio, neural state, zone state, backups
reporting.py	726	Reporting	P&L; summaries, hourly reports, zone transition alerts, session statistics
user_manager.py	373	Infrastructure	UserManager: active user loading, multi-user database operations
pivot_analysis.py	288	Infrastructure	ATH-based zone boundaries; 5-Unit pool fractions; zone info API
main.py	1,164	Orchestration	Startup sequence, license check, signal init, watchdog, concurrent trading
run.py	518	Entry Point	CLI parsing, pre-checks, neural DB status, installation verification
TOTAL — 19 modules	19,994+		Est. 700–1,100+ hours specialist development

9 Software Valuation

The valuation is based on three independent methods: (1) replacement cost estimation, calculated from labour hours at specialist developer rates; (2) market comparables, benchmarked against publicly available algorithmic trading software of comparable scope; and (3) commercial licensing precedent derived from industry norms for proprietary fintech software.

Comparable	Value Range	Notes
Simple single-strategy crypto bot	$100 – $2,000	Basic infrastructure, single asset, no state management
Mid-tier Python bot (single strategy)	$2,000 – $15,000	Custom builds, limited position management
Multi-strategy Python bot (3+ strategies)	$15,000 – $50,000	Professional builds, partial infrastructure
HVTS-Liquidity Protocol (this system)	$35,000 – $90,000	7 strategies, per-symbol BiLSTM neural, multi-user, full infra
HVTS-Engine1 (companion MT5 Forex system)	$40,000 – $80,000	15 strategies, Triple-Signal Consensus, 9-Stage Deep Scanner
Institutional-grade equivalent	$150,000 – $500,000+	Full production with SLA and team maintenance

$60K–$120K

Replacement Cost Rebuild from scratch

$35K–$75K

Private Sale Value Source code + IP transfer

$200–$600

SaaS / Month Per subscriber, recurring

10 Development Cost Breakdown

The following table itemises estimated development hours by work category at a representative senior quantitative developer rate of $150/hr — a conservative estimate for specialist Python + CCXT + exchange API + deep learning + algorithmic finance expertise.

Work Category	Est. Hours	Est. Cost @ $150/hr
Strategy research & algorithm design (7 strategies)	120 – 180 hrs	$18,000 – $27,000
7 strategy module implementations	200 – 300 hrs	$30,000 – $45,000
Triple OR Signal Protocol & consensus engine	60 – 90 hrs	$9,000 – $13,500
Per-symbol neural signal system (11+ databases)	60 – 90 hrs	$9,000 – $13,500
BiLSTM-Attention neural network architecture	50 – 80 hrs	$7,500 – $12,000
24-feature engineering & data pipeline	30 – 50 hrs	$4,500 – $7,500
Circuit breaker, synthetic fallback, walk-forward CV	25 – 40 hrs	$3,750 – $6,000
GMMA multi-timeframe indicator (H1/H4/Daily)	30 – 50 hrs	$4,500 – $7,500
SuperTrend Oracle integration (optional overlay)	50 – 80 hrs	$7,500 – $12,000
ATH-based 5-Unit Zone Fund Allocation System	30 – 50 hrs	$4,500 – $7,500
Multi-user configuration & isolation layer	50 – 80 hrs	$7,500 – $12,000
Gate.io CCXT async integration & exchange abstraction	50 – 80 hrs	$7,500 – $12,000
SQLite state management & persistence layer	40 – 60 hrs	$6,000 – $9,000
Node-locked license system + activation workflow	25 – 40 hrs	$3,750 – $6,000
Telegram & reporting system (all event coverage)	30 – 50 hrs	$4,500 – $7,500
Testing, debugging & hardening (19 modules)	80 – 150 hrs	$12,000 – $22,500
Configuration, DevOps & run.py CLI	20 – 40 hrs	$3,000 – $6,000
TOTAL — Senior Quant Developer @ $150/hr	900 – 1,420 hrs	$135,000 – $213,000

11 Commercial Licensing Scenarios

Licensing Scenario	Value Range	Notes
Private source code sale (full IP transfer)	$35,000 – $75,000	Buyer receives full ownership; seller retains no rights
White-label / rebranded sale	$20,000 – $50,000	Buyer rebrands; seller may retain re-licensing rights
SaaS subscription (hosted and managed)	$200 – $600 / month	Per-seat recurring; LicenseManager supports machine-lock
Individual strategy module sale	$3,000 – $12,000 each	7 strategy modules available for individual licensing
Crypto exchange / white-label provider	$40,000 – $120,000	Exchange-wide licence; Gate.io-native design simplifies integration
Prop trading firm / fund licence	$50,000 – $150,000	Firm-wide licence plus royalty on AUM or profits
Bundle: Liquidity Protocol + Engine1 (Forex)	$60,000 – $130,000	Cross-asset bundle: crypto spot (Gate.io) + MT5 Forex
Ongoing support & maintenance retainer	$1,500 – $4,000/month	SLA-backed; updates, monitoring, new asset onboarding
White-label + hosted SaaS platform build-out	$60,000 – $140,000	Full product launch with branding, billing layer, and infrastructure

12 Target Buyers & Use Cases

Independent Trader / Algo Developer

Acquires a battle-tested 7-strategy multi-asset system supporting 11+ cryptocurrency pairs simultaneously, bypassing 12–18 months of development. The per-symbol BiLSTM-Attention neural signal architecture and Triple OR Protocol enable immediate deployment across a diversified spot portfolio on Gate.io.

Crypto SaaS Startup

White-labels the system as a multi-asset auto-trading subscription product. The built-in multi-user configuration layer and LicenseManager handle per-subscriber isolation and machine-locked access control out of the box.

Gate.io Power User / High-Net-Worth Trader

Deploys HVTS-Liquidity Protocol on a single Gate.io account to run 7 concurrent strategies across 11 assets simultaneously, maximising capital utilisation through the 5-Unit Zone Allocation System.

Prop Trading Firm

Licences the system to run on internal funded Gate.io accounts. The seven-strategy diversification and per-symbol neural signals provide independent alpha across the crypto spot market.

Crypto Fund / Family Office

Uses the BiLSTM-Attention Neural Network outputs, GMMA multi-timeframe overlays, and ATH zone system as systematic entry signals on discretionary multi-asset positions.

Software Broker / Reseller

Acquires the source code and resells individual strategy modules or bundled packages. The seven independently configurable strategies provide multiple product SKUs.

13 Risk Considerations

Market Risk

Algorithmic strategies do not guarantee profit. Cryptocurrency spot markets are highly volatile. Past simulated or live performance does not guarantee future results. All seven strategies can incur simultaneous losses during sustained bearish markets.

Model Risk

The neural network is trained on historical data and cannot predict black-swan events or structural market regime changes. The 12-hour retraining cycle mitigates but does not eliminate model staleness.

Execution Risk

Order execution quality depends on Gate.io infrastructure, network latency, and API stability. Slippage, partial fills, and API rate-limit rejections can materially affect strategy outcomes.

Dependency Risk

Requires a live Gate.io API connection, reliable internet, Python 3.8+ environment, TensorFlow 2.12+, and access to per-symbol neural databases.

Strategy Risk

DCA, Accumulator, and Pivot strategies increase exposure when price trends strongly against open positions.

Multi-Asset Correlation Risk

During broad crypto market corrections, all 11+ traded assets may decline simultaneously.

License Risk

The node-locked license system ties the software to a specific machine ID. Operators must plan for license re-issuance on hardware changes.

Technology Risk

Provided as source code without warranty. The buyer is responsible for infrastructure, deployment security, and ongoing maintenance.

Regulatory Risk

Automated trading regulations vary by jurisdiction. The buyer is solely responsible for ensuring compliance with applicable laws and Gate.io terms of service.

SOLUTION 2 — HVTS-FUTURES FUND MANAGER

Quadruple Signal Protocol — Neural + GMMA 1D/4H/1H — Long-Only Flip-Based DCA on Gate.io Futures

14 HVTS-Futures Fund Manager — Executive Summary

1,229	4	90+	5	10x
Lines of Code Single focused module	Signal Sources Neural + GMMA 1D/4H/1H	Supported Symbols Gate.io Perpetual Futures	Max DCA Stages Configurable per session	Default Leverage Configurable 1x–100x

HVTS-Futures Fund Manager is a focused, production-grade perpetual futures trading engine built on Gate.io's futures market. Where HVTS-Liquidity Protocol governs seven concurrent spot strategies across eleven assets, the Futures Fund Manager distils the signal consensus into a single precision instrument: a Quadruple Signal Protocol governing a long-only, flip-triggered DCA system across 90+ USDT perpetual futures pairs.

The system combines Supabase-hosted neural network predictions with three independent GMMA timeframes (1D, 4H, 1H) to enforce a strict all-or-nothing entry gate — all four signals must be simultaneously BULLISH before a new DCA stage opens. A single signal turning non-BULLISH triggers an immediate emergency exit across all open stages. This asymmetric discipline — maximum conviction on entry, zero tolerance on exit — defines the Futures Fund Manager's risk philosophy.

Product	HVTS-Futures Fund Manager — Quadruple Signal Flip-DCA
Asset Class	Cryptocurrency — Perpetual Futures (USDT-margined, Gate.io)
Platform	Gate.io Futures — Python CCXT Async API
Signal Protocol	Quadruple: Neural (Supabase) + GMMA 1D + GMMA 4H + GMMA 1H
Strategy Type	Long-Only, Flip-Triggered DCA (up to 5 stages per symbol)
Symbol Coverage	90+ USDT perpetual futures pairs (Tier 1 + Tier 2 + Tier 3)
Entry Condition	ALL FOUR signals BULLISH simultaneously
Exit Condition	ANY signal turns non-BULLISH → emergency exit all stages immediately
Leverage	Configurable 1x–100x (default 10x); uniform across all entries
Position Sizing	1% margin per stage (configurable) or fixed USDT; max 50% total margin
DCA Trigger	Neural "flip" to BULLISH (new bullish edge crossing) opens next DCA stage
Max DCA Stages	5 per symbol (configurable via --max-dca)
Stop Loss	Configurable % loss trigger (default -50% per position)
Neural Source	Supabase cloud DB — real-time BiLSTM predictions per symbol

GMMA Engine	Built-in GMMA class; crypto-optimised EMA periods; lazy refresh per timeframe
State Persistence	SQLite (trading_state.db + trading.db) — survives restarts
License	Node-locked via LicenseManager; machine ID validation on startup
Codebase	1,229 lines (hvts_ai_fund_manager.py) + exchange.py + license_manager.py
Entry Point	python hvts_ai_fund_manager.py [--dry-run] [--symbols BTC,ETH] [--max-dca 5]

15 Quadruple Signal Protocol

The Futures Fund Manager replaces the Spot system's Triple OR Protocol with a stricter Quadruple AND Protocol. Where the spot system allows any one of three signal pairs to authorise entry (maximising opportunity), the futures system requires all four signals to agree simultaneously — a deliberate design choice reflecting the amplified risk of leveraged futures positions.

Signal Sources

Signal Source	Module	Timeframe	Description
Neural Network (Supabase)	SupabaseNeuralReader	External DB	BiLSTM predictions per symbol; 5-class trend mapped to BULLISH/BEARISH/NEUTRAL
GMMA Daily	GMMAUpdater + GMMADatabaseReader	1D	Guppy Multiple Moving Average on daily bars; refresh threshold 24h
GMMA 4-Hour	GMMAUpdater + GMMADatabaseReader	4H	GMMA on 4-hour bars; refresh threshold 4h; crypto EMA periods
GMMA 1-Hour	GMMAUpdater + GMMADatabaseReader	1H	GMMA on 1-hour bars; refresh threshold 1h; fastest-responding signal

Consensus Gate Rules

Condition	Rule	Action
ENTRY	Neural BULLISH AND GMMA 1D BULLISH AND GMMA 4H BULLISH AND GMMA 1H BULLISH	Open next DCA stage on new neural flip
EMERGENCY EXIT	ANY one of the four signals turns non-BULLISH	Close ALL open stages immediately at market
HOLD	All signals bullish but no new neural flip since last acted	No action — hold existing positions, no new entries
WAITING	Not all signals bullish, no open positions	Monitor — wait for full consensus to reform
STOP LOSS	Individual position loss exceeds MAX_LOSS_PERCENT	Close that position regardless of signal state

Neural Flip Mechanism

The entry trigger is not simply "neural is bullish" but specifically a neural flip to bullish — the moment the neural signal transitions from non-BULLISH to BULLISH while all three GMMA timeframes confirm. Each flip increments a persistent flip counter per symbol. The system tracks the last acted flip count separately, so missed flips during downtime are detected and replayed on the next startup cycle.

Flip State Field	Description
flip_count	Total number of bullish flips detected for this symbol since last reset
last_acted_flip_count	Flip count at which the most recent DCA stage was opened
Unacted flip (flip_count > last_acted)	Triggers opening of next DCA stage (new_stage = last_acted + 1)
Startup replay	If neural is BULLISH on startup but last_seen was non-BULLISH, increments flip_count and replays

Flip State Field	Description
GMMA regime change	If GMMA daily regime changes, flip_count resets to 0 — prevents stale flips carrying over
Max stages reached	If new_stage > MAX_DCA_ENTRIES, flip is acknowledged but no position opened

16 Built-In GMMA Engine

Unlike the spot system which delegates GMMA computation to the gmma_manager module, the Futures Fund Manager includes a self-contained GMMA class. This eliminates an external dependency and allows the futures bot to run as a standalone binary without the full 19-module spot codebase.

Period Set	Crypto-optimised: Short [4,7,10,13,16,20], Long [30,36,42,48,54,60]
Computation	EMA (exponential smoothing, adjust=False) for each of 12 periods
Signal Classification	short_avg > long_avg → bullish (normalised spread >0.5% = strong_bullish)
Trend Score	50 + spread-weighted score (0–100 scale); used for debug output
Minimum Bars	70 bars required; returns neutral if insufficient data
Refresh Thresholds	1D: 86,400s · 4H: 14,400s · 1H: 3,600s — lazy refresh avoids excessive API calls
Storage	GMMASignals table in trading_state.db; INSERT OR REPLACE per symbol+timeframe
Cache Layer	In-memory dict cache keyed (symbol, timeframe); checks threshold before DB read

17 Long-Only DCA Position Management

The Futures Fund Manager is a strictly long-only system. Short positions are never opened. Each symbol can hold up to MAX_DCA_ENTRIES concurrent long positions, each representing one DCA stage. Stages are opened sequentially on each new bullish neural flip, and all stages are closed together on any signal disagreement.

Position Sizing

Sizing Rule	Description
Fixed Mode (FIXED_POSITION_SIZE env)	Opens each DCA stage with a fixed USDT margin regardless of balance
Percentage Mode (default)	Base margin = free_balance × 1%; respects max 50% total margin cap
Equal sizing (DCA_SIZING_MODE=equal)	All stages use the same base margin
Increasing sizing	Each stage uses base × (1 + (stage-1) × 0.5) — larger later entries
Decreasing sizing	Each stage uses base / (1 + (stage-1) × 0.5) — smaller later entries
Leverage	Uniform across all stages; configurable 1x–100x (default 10x via LEVERAGE env var)
Notional size	margin_usdt × leverage — submitted to Gate.io as the order size
Minimum balance	If free balance < $50 USDT (or FIXED_POSITION_SIZE), no new positions opened

Position Lifecycle

Stage	Trigger	Action
1 — Signal Consensus	All four signals bullish	System enters "ready" state for this symbol
2 — Neural Flip Detected	flip_count > last_acted_flip_count	New DCA stage authorised
3 — Margin Calculation	calculate_dca_margin()	Checks balance, cap, sizing mode
4 — Order Placement	gate_execute_trade(buy)	Market order via Gate.io futures API
5 — Position Saved	PositionStore.save()	Written to neural_flip_positions table
6 — Flip Counter Updated	update_last_acted_flip()	Prevents duplicate entries on same flip
7 — Monitor	process_symbol() every CHECK_INTERVAL	Checks signals and stop loss each cycle
8 — Exit Trigger	Any signal non-BULLISH OR stop loss hit	Close position via gate_execute_trade(sell, reduce_only=True)
9 — Position Closed	PositionStore.close()	Records exit price, P&L;, reason in DB

18 Symbol Coverage — 90+ Gate.io Perpetual Futures

HVTS-Futures Fund Manager ships with a hardcoded SYMBOL_MAP of 90+ Gate.io perpetual futures pairs organised across three liquidity tiers. The system scans all symbols on each cycle, opening positions only where all four signals align. Symbols can be overridden at startup with --symbols BTC,ETH,SOL.

Tier	Symbols	Characteristics
Tier 1 — Ultra High Volume (10)	BTC, ETH, SOL, XRP, DOGE, BNB, ADA, AVAX, LTC, LINK	Deepest liquidity; tightest spreads; highest neural signal confidence
Tier 2 — High Volume Majors (20)	XLM, BCH, NEAR, UNI, TRX, SHIB, WLD, ICP, DOT, INJ, FIL, ZEC, AAVE, ARB, FET, RUNE, XMR, CRV, GALA, CHZ	Established mid-cap futures; daily volume >$50M USDT
Tier 3 — Mid-Cap Liquid Futures (60+)	ENJ, ETC, XTZ, THETA, EOS, NEO, DASH, KSM, CELO, IOTA, FLOW, RENDER, SAND, MANA, AXS, GRT, SNX, TAO, JTO, ONDO, ENA, SEI, SUI, PEPE, TRUMP, PI, VIRTUAL, HYPE, ORDI, JUP, and 30+ more	Emerging assets; filtered by Gate.io futures availability

The gate_futures_volume_filter.py utility (included in the package) provides on-demand volume screening: operators can pass --min-volume to filter the active symbol list to only those with >N USDT daily volume, ensuring the bot trades only liquid markets regardless of the default symbol map.

19 Architecture & Infrastructure

Module Structure

Module	Lines	Category	Primary Responsibility
hvts_ai_fund_manager.py	1,229	Core	Main bot: GMMA engine, Supabase reader, DCA logic, position management, signal loop
exchange.py	~400	Infrastructure	Gate.io futures connector: gate_execute_trade(), get_gate_connection(), gate_fetch_ohlcv()
license_manager.py	847	Infrastructure	Node-locked license validation; machine ID; expiry; hard abort on failure
gate_futures_volume_filter.py	~200	Utility	Gate.io volume screener; filters symbol list by daily USDT volume threshold
build_exe.py	~300	DevOps	PyInstaller build script; packages all modules into single Windows/Linux executable
generate_licenses.py	~250	Admin	License key generation tool (admin only); batch CSV support; machine-locked or open keys

Database Architecture

trading_state.db	GMMASignals table: stores computed GMMA per symbol/timeframe with full indicator metadata; neural_signal_state table: persistent flip counter, last acted count, GMMA regime per symbol
trading.db	neural_flip_positions table: all open and closed DCA positions with entry/exit price, quantity, leverage, stage, status, P&L;, and close reason

Supabase Neural Integration

The Futures Fund Manager reads neural signals from a Supabase-hosted PostgreSQL table (neural_signals) populated by the HVTS-NeuralNetwork engine. The SupabaseNeuralReader class queries the most recent record per symbol, applies a 30-second in-memory cache to respect rate limits, and maps the raw 5-class trend output to a binary BULLISH/BEARISH/NEUTRAL signal. Bearish signals are deliberately mapped to NEUTRAL in long-only mode — preventing false negatives from triggering unnecessary exits when the raw model output is mildly negative.

Table	neural_signals (Supabase PostgreSQL)
Key fields consumed	trend, current_price, model_confidence, target_price_1, expected_return_1h, risk_level, signal_strength, created_at
Trend mapping	STRONG_BULL / BULL → BULLISH; STRONG_BEAR / BEAR → BEARISH (then → NEUTRAL in long-only); else NEUTRAL
Cache TTL	30 seconds — balances API cost against signal freshness
Confidence field	Passed through to signal_data but not used as entry filter in long-only mode (gate is signal state only)
Fallback	Returns None if Supabase unreachable; NEUTRAL assumed; no position opened

20 CLI Interface & Configuration

Command-Line Arguments

Argument	Description
--dry-run	Simulation mode — no real orders placed; balance returns $1,000 USDT; all logic executes normally
--debug	Enables DEBUG-level logging output
--symbols BTC,ETH,SOL	Override symbol list (slash format auto-added if missing; must exist in SYMBOL_MAP)
--interval N	Check interval in seconds between processing cycles (default: 60)
--max-dca N	Maximum DCA stages per symbol (default: 5)
--dca-mode equal|increasing|decreasing	Position sizing progression across DCA stages
--show-machine-id	Print this machine's license ID and exit
--status	Show current license status and exit
--activate [KEY]	Activate a license key; prompts if KEY omitted

Environment Variables (.env)

GATE_API_KEY	Gate.io API key (required for live trading)
GATE_SECRET	Gate.io API secret (required for live trading)
FIXED_POSITION_SIZE	Fixed USDT margin per DCA stage (overrides percentage mode if set)
LEVERAGE	Leverage multiplier 1–100 (default: 10)
MAX_DCA_ENTRIES	Maximum DCA stages per symbol (default: 5)
DCA_SIZING_MODE	equal | increasing | decreasing (default: equal)
CHECK_INTERVAL	Seconds between processing cycles (default: 60)
MAX_LOSS_PERCENT	Stop loss threshold as negative percentage (default: -50.0)

21 HVTS-Futures Fund Manager — Valuation

The Futures Fund Manager is priced as a standalone product, as a bundle with the Liquidity Protocol, or as a SaaS-deployable futures signal service. Its value derives from the Quadruple Signal Protocol, flip-triggered DCA architecture, Supabase neural integration, and Gate.io futures-native execution — all validated in production across 90+ futures pairs.

Licensing Scenario	Value Range	Notes
Standalone source code sale	$15,000 – $30,000	Full ownership of hvts_ai_fund_manager.py + exchange.py
Bundle: Futures + Liquidity Protocol	$45,000 – $95,000	Both systems; cross-asset coverage spot + futures
White-label futures SaaS	$200 – $500/month	Per subscriber; LicenseManager handles machine-lock
Prop firm / fund licence	$25,000 – $60,000	Firm-wide; royalty on AUM optional
API-as-a-service (signals only)	$100 – $300/month	Supabase neural signals + GMMA computation layer
Ongoing maintenance retainer	$1,000 – $2,500/month	Updates, new symbol onboarding, signal tuning

$15K–$30K	$45K–$95K	$200–$500
Standalone Sale Futures module only	Bundle Value Spot + Futures systems	SaaS / Month Per subscriber

22 Comparison: Liquidity Protocol vs Futures Fund Manager

Attribute	HVTS-Liquidity Protocol (Spot)	HVTS-Futures Fund Manager
Market Type	Gate.io Spot	Gate.io Perpetual Futures
Directions	Long only (spot buy)	Long only (with leverage)
Signal Protocol	Triple OR (any one pair sufficient)	Quadruple AND (all four required)
Signal Sources	Neural + GMMA Daily/H4/H1 (OR pairs)	Neural + GMMA 1D + GMMA 4H + GMMA 1H (AND)
Neural Source	Per-symbol SQLite DB (local)	Supabase PostgreSQL (cloud-hosted)
Entry Trigger	Any OR pair consensus	Neural flip to BULLISH while all GMMA bullish
Exit Trigger	GMMA Daily bearish (emergency) / ALL bearish (force)	ANY signal non-BULLISH (immediate)
Strategies	7 concurrent (DCA, Grid, AMM, Signal, Scalper, Pivot, Accumulator)	1 (Long-Only Flip-DCA)
Leverage	None (spot)	1x–100x (default 10x)
Symbol Count	11+ spot pairs	90+ perpetual futures pairs
Codebase Size	19,994+ lines (19 modules)	~1,229 lines (1 main module + utils)
Multi-User	Yes — full per-user isolation	No — single operator per instance
Zone System	Yes — 5-Unit ATH-based allocation	No — margin-percentage based
Stop Loss	Force-close on all-bearish consensus	Configurable % loss per position (MAX_LOSS_PERCENT)
Complexity	High — enterprise-grade infrastructure	Focused — production-ready in fewer dependencies
Deployment	Full Python environment + 19 modules	Single executable (PyInstaller) or Python script

23 Summary & Next Steps

✓ 19,994+ lines of production Python across 19 uploaded modules — estimated 900–1,420+ hours of specialist development time.

✓ Seven concurrent strategies: DCA, Grid, AMM, Signal, Scalper, Pivot, Accumulator — each independently configurable and deployable per user.

✓ HVTS-NeuralNetwork — Dual-Branch Deep Learning Engine: BiLSTM + Multi-Head Attention (4 heads) + Residual Dense — Keras Functional API producing four simultaneous outputs.

✓ 24-feature technical indicator space — RSI, MACD, ATR, Bollinger-derived features, moving average deviations, volume ratios, and trend strength.

✓ Triple OR Signal Protocol: (Neural + GMMA Daily) OR (GMMA H4 + GMMA Daily) OR (GMMA H1 + GMMA Daily) — asymmetric entry with strict exit discipline.

✓ Per-symbol neural signals: each of 11+ trading pairs reads from its own dedicated SQLite database with NEUTRAL-state protection.

✓ 5-Unit ATH-Based Zone Allocation: all-in capital divided into five 20% pools mapped to ATH discount zones — all zones tradeable.

✓ Multi-user architecture: full per-user isolation for exchange credentials, symbols, strategies, AMM parameters, zone overrides.

✓ Force-close protection: ALL FOUR signals bearish simultaneously triggers a full position close and pending order cancellation.

✓ Node-locked license system: LicenseManager validates machine ID on startup; activation workflow supports single and batch key issuance.

✓ Enterprise infrastructure: Gate.io CCXT async native, SQLite WAL persistence, automated backups, watchdog, credential validation guard.

HVTS-FUTURES FUND MANAGER — Additional Solution:

✓ Quadruple Signal Protocol (Neural AND GMMA 1D AND GMMA 4H AND GMMA 1H) — stricter than spot Triple OR; designed for leveraged futures discipline.

✓ 90+ Gate.io perpetual futures pairs across three liquidity tiers — Tier 1 ultra-high volume through Tier 3 mid-cap liquid futures.

✓ Long-only flip-triggered DCA: up to 5 DCA stages per symbol; each stage opened on a new neural bullish flip with full GMMA confirmation.

✓ Supabase-integrated neural signals: cloud-hosted BiLSTM predictions consumed in real-time with 30-second cache and automatic NEUTRAL fallback.

✓ Single-module deployment: 1,229-line self-contained bot; ships as a PyInstaller executable; no 19-module dependency chain required.

✓ Emergency exit on any signal disagreement: zero tolerance exit policy reflects leveraged risk profile.

✓ Combined offering: Liquidity Protocol (spot) + Futures Fund Manager = cross-market coverage of Gate.io's entire USDT ecosystem.

✓ Replacement cost $60,000–$120,000 (Liquidity Protocol) + $15,000–$30,000 (Futures Fund Manager). Bundle private sale value $45,000–$95,000.

✓ SaaS potential: $200–$600/month (spot subscribers) + $200–$500/month (futures subscribers).

To arrange a technical demonstration, request full codebase access for due diligence, or discuss licensing terms — contact the HVTS-Liquidity Protocol development team.

Disclaimer: This document is for informational purposes only and does not constitute financial advice, a prospectus under securities law, or an offer to sell a financial product. Algorithmic trading involves significant risk of loss. Past performance is not indicative of future results. All valuations are estimates based on market comparables and labour cost analysis and do not constitute a formal appraisal.

SOLUTION 3 — HVTS-AI-FOREX

Neural-GMMA Adaptive Forex Engine

Quadruple Signal Protocol · MT5 Native · 28 Major & Minor Pairs · Adaptive Position Sizing · Multi-Session Awareness

24

HVTS-AI-FOREX — Executive Summary

3,840+	28	4	5	1:500
Lines of Code	Forex Pairs Majors & Minors	Signal Sources Neural+GMMA 1D/4H/1H	DCA Stages per symbol max	Max Leverage Configurable

HVTS-AI-FOREX is a production-grade, neural-augmented algorithmic Forex trading engine purpose-built for MetaTrader 5 (MT5). It applies the same proven Quadruple Signal Protocol used in the HVTS-Futures Fund Manager — requiring simultaneous bullish alignment across a BiLSTM neural network and three GMMA timeframes (Daily, H4, H1) — to the 28 most liquid major and minor currency pairs on the global Forex market.

Unlike the crypto spot and futures systems, HVTS-AI-FOREX is engineered for the unique characteristics of foreign exchange: 24/5 continuous sessions, tight institutional spreads, pip-denominated P&L, multi-currency account management, and broker-specific execution constraints. The system integrates directly with MT5 via the `MetaTrader5` Python package, executing live market and pending orders through any MT5-compatible broker.

Attribute	Value
Trading Assets	28 Forex pairs: 7 majors (EUR/USD, GBP/USD, USD/JPY, USD/CHF, AUD/USD, USD/CAD, NZD/USD) + 21 minors
Platform	MetaTrader 5 — Python MT5 API (MetaTrader5 package)
Signal Protocol	Quadruple AND: Neural (BiLSTM) + GMMA Daily + GMMA H4 + GMMA H1
Neural Source	Supabase PostgreSQL — real-time BiLSTM predictions per symbol
Entry Condition	ALL FOUR signals BULLISH simultaneously + session filter active
Exit Condition	ANY signal turns non-BULLISH → emergency close all stages at market
Strategy	Long-only flip-triggered DCA (up to 5 stages per symbol)
Leverage	Configurable 1:1 – 1:500; default 1:100 (broker-dependent)
Session Filter	London + New York overlap priority; configurable session windows
Position Sizing	ATR-normalised lot sizing; max 2% account risk per stage
Stop Loss	ATR-based dynamic SL + hard pip floor; configurable multiplier
Take Profit	Configurable R:R ratio (default 2:1); trailing stop optional
State Persistence	SQLite (forex_state.db) — survives MT5 restarts and reconnections

Attribute	Value
License	Node-locked via LicenseManager; machine ID validation on startup
Codebase	3,840+ lines across 6 modules + shared license_manager.py

25

Quadruple Signal Protocol — Forex Edition

The Forex edition inherits the Quadruple AND gate from the Futures Fund Manager and extends it with two Forex-specific enhancements: a **session activity filter** that suppresses entries outside high-liquidity windows, and a **spread guard** that blocks entries when live broker spread exceeds a configurable pip threshold.

Signal Source	Module	Timeframe	Description
Neural Network (Supabase)	SupabaseForexReader	External DB	BiLSTM predictions per pair; 5-class trend → BULLISH/BEARISH/NEUTRAL
GMMA Daily	ForexGMMAUpdater	D1	Guppy MMA on daily bars; primary trend gate; refresh 24 h
GMMA H4	ForexGMMAUpdater	H4	GMMA on 4-hour bars; intermediate momentum; refresh 4 h
GMMA H1	ForexGMMAUpdater	H1	GMMA on 1-hour bars; intraday trigger; refresh 1 h
Session Filter	SessionManager	Realtime	London/NY overlap window; suppresses entries in illiquid hours
Spread Guard	MT5Connector	Tick	Live spread check vs MAX_SPREAD_PIPS; blocks high-spread entries

Consensus Gate Rules

Condition	Rule	Action
ENTRY	Neural BULLISH AND GMMA D1 BULLISH AND GMMA H4 BULLISH AND GMMA H1 BULLISH AND session active AND spread ≤ MAX_SPREAD_PIPS	Open next DCA stage on new neural flip
EMERGENCY EXIT	ANY one of the four signals turns non-BULLISH	Close ALL open stages at market immediately
HOLD	All signals bullish but no new neural flip since last acted	No action — hold existing positions
SPREAD BLOCK	Live spread > MAX_SPREAD_PIPS (e.g. news events)	Skip entry; log warning; retry next cycle
SESSION BLOCK	Outside configured active session window	Suppress entry; existing positions held and monitored
STOP LOSS	Position loss > ATR-based SL or hard pip floor breached	Close that position regardless of signal state

26

Architecture — 6-Module Design

HVTS-AI-FOREX is structured as a focused 6-module system. Unlike the 19-module Liquidity Protocol, the Forex engine prioritises portability and single-broker deployment, packaging the complete trading loop — signal consumption, GMMA computation, MT5 execution, position management, and state persistence — into a lean, auditable codebase.

Module	Lines	Category	Primary Responsibility
hvts_ai_forex.py	1,680	Core	Main trading loop: signal orchestration, DCA logic, position lifecycle, session management, spread guard
mt5_connector.py	520	Infrastructure	MT5 connection manager: order placement, position query, tick fetch, OHLCV retrieval, reconnection handler
forex_gmma.py	340	Signal Engine	Self-contained GMMA engine: crypto-period EMA sets adapted for FX, lazy refresh per timeframe, SQLite cache
supabase_forex_reader.py	290	Signal Engine	Supabase PostgreSQL reader: BiLSTM predictions per pair, 30s cache TTL, NEUTRAL fallback on error
forex_state.py	560	Infrastructure	SQLite state manager: open/closed positions, flip counters, GMMA state, P&L history, WAL mode
license_manager.py	847	Infrastructure	Shared node-locked license: machine ID validation, expiry, hard abort + log on failure (shared with other HVTS products)
TOTAL — 6 modules	3,840+	—	Est. 200–320 hours specialist development

27

Pair Coverage — 28 Major & Minor Forex Pairs

HVTS-AI-FOREX ships with a hardcoded SYMBOL_MAP of 28 currency pairs validated against MT5 broker symbol naming conventions. The system scans all pairs on each cycle, opening positions only where all four signals and session/spread filters align. Symbols can be overridden at startup with `--symbols EURUSD,GBPUSD`.

Tier	Pairs	Characteristics
Tier 1 — G7 Majors (7)	EUR/USD, GBP/USD, USD/JPY, USD/CHF, AUD/USD, USD/CAD, NZD/USD	Tightest spreads (0.1–1.5 pips); deepest global liquidity; highest neural signal confidence; London + NY overlap prioritised
Tier 2 — EUR & GBP Crosses (11)	EUR/GBP, EUR/JPY, EUR/CHF, EUR/AUD, EUR/CAD, EUR/NZD, GBP/JPY, GBP/CHF, GBP/AUD, GBP/CAD, GBP/NZD	Moderate spreads (1–4 pips); strong institutional flow; London session primary; daily volume >$50B USD equivalent
Tier 3 — Commodity & Asia Crosses (10)	AUD/JPY, AUD/NZD, AUD/CAD, AUD/CHF, NZD/JPY, NZD/CAD, NZD/CHF, CAD/JPY, CHF/JPY, USD/NOK	Wider spreads (2–8 pips); session-sensitive entries; spread guard critical; Asia + Sydney session primary

28

Adaptive Position Management

The Forex engine replaces the crypto margin-percentage sizing model with ATR-normalised lot sizing — a standard institutional Forex risk framework. Each DCA stage risks a fixed percentage of account equity, with lot size calculated dynamically from the current ATR(14) on H1 bars and the pair's pip value.

Position Sizing Formula

Lot Size = (Account Equity × Risk%) ÷ (ATR Multiplier × ATR × Pip Value)

Parameter	Default	Description
RISK_PER_STAGE_PCT	1.0%	Percentage of account equity risked per DCA stage
ATR_PERIOD	14	ATR lookback on H1 bars; re-computed on each entry
ATR_SL_MULTIPLIER	1.5×	Stop loss = ATR × multiplier in pips; dynamic per entry
MIN_SL_PIPS	10 pips	Hard floor stop loss; overrides ATR if ATR is smaller
TP_RR_RATIO	2.0	Take profit = SL distance × R:R ratio

Parameter	Default	Description
TRAILING_STOP	Optional	Activates trailing stop at 1× ATR profit; trails at 0.5× ATR
MAX_STAGES	5	Maximum concurrent DCA stages per pair
MAX_ACCOUNT_RISK_PCT	10%	Hard cap: total open risk across all pairs and stages

29

Multi-Session Awareness

Forex liquidity and volatility vary dramatically across the 24-hour trading day. HVTS-AI-FOREX implements a SessionManager that tracks the four major Forex sessions and enforces session-specific entry rules to avoid low-liquidity periods (Sunday open, Friday close, and inter-session gaps).

Session	UTC Window	Priority Pairs	Entry Rule
Sydney	21:00 – 06:00	AUD/*, NZD/*, JPY crosses	Tier 3 pairs only; reduced lot size (50% of base)
Tokyo / Asia	00:00 – 09:00	JPY pairs, AUD/*, NZD/*	Full entries permitted for Tier 1 JPY + Tier 3 Asia pairs
London	07:00 – 16:00	EUR/*, GBP/*, CHF/*	Full entries all tiers; highest liquidity window
London + NY Overlap	12:00 – 16:00	All 28 pairs	Maximum activity window; all entries permitted; full lot size
New York	13:00 – 22:00	USD/*, CAD/*, all majors	Full entries Tier 1 + Tier 2; Tier 3 reduced lot
Dead Zone	20:00 – 21:00	—	All entries suppressed; existing positions held

30

CLI Interface & Configuration

Argument	Description
--dry-run	Simulation mode — no real MT5 orders placed; virtual $10,000 balance; all signal logic executes normally
--debug	Enables DEBUG-level logging; prints raw GMMA values and signal states per cycle
--symbols EURUSD,GBPUSD	Override default 28-pair symbol list; must exist in SYMBOL_MAP and on broker
--interval N	Check interval in seconds between processing cycles (default: 60)
--max-dca N	Maximum DCA stages per symbol (default: 5)
--dca-mode equal|inc|dec	Position sizing mode across DCA stages: equal / increasing / decreasing
--session london|ny|asia|all	Override session filter; "all" disables session gating (use with caution)
--show-machine-id	Print this machine's license ID and exit
--activate [KEY]	Activate a license key; prompts if KEY omitted

Environment Variables (.env)

Variable	Description
MT5_LOGIN	MT5 account number (integer)
MT5_PASSWORD	MT5 account password
MT5_SERVER	Broker MT5 server name (e.g. BrokerName-Live)
SUPABASE_URL	Supabase project URL for neural signal reads

Variable	Description
SUPABASE_KEY	Supabase anon/service key
RISK_PER_STAGE_PCT	Account risk % per DCA stage (default: 1.0)
MAX_ACCOUNT_RISK_PCT	Hard cap total open risk % (default: 10.0)
MAX_SPREAD_PIPS	Maximum spread to allow entry in pips (default: 5.0)
ATR_SL_MULTIPLIER	ATR multiplier for stop loss (default: 1.5)
TP_RR_RATIO	Take profit R:R ratio (default: 2.0)
LEVERAGE	Account leverage 1–500 (default: 100; broker-constrained)
CHECK_INTERVAL	Seconds between processing cycles (default: 60)
TRAILING_STOP_ENABLED	Enable trailing stop: true/false (default: false)

31

HVTS-AI-FOREX — Valuation

HVTS-AI-FOREX is priced as a standalone Forex product, as a bundle with either or both HVTS systems, or as a SaaS-deployable Forex signal service. Its value derives from the Quadruple Signal Protocol adapted for FX, ATR-normalised adaptive position sizing, multi-session awareness, MT5-native execution across 28 pairs, and shared neural infrastructure with the existing HVTS ecosystem.

Licensing Scenario	Value Range	Notes
Standalone source code sale	$18,000 – $35,000	Full ownership of 6 Forex modules; MT5-native; 28-pair coverage
Bundle: Forex + Futures Fund Manager	$28,000 – $55,000	Cross-asset long-DCA: FX 28 pairs + Gate.io 90+ futures
Bundle: All Three HVTS Systems	$55,000 – $110,000	Complete suite: spot crypto + perpetual futures + Forex
White-label Forex SaaS	$200 – $500/month	Per subscriber; LicenseManager handles machine-lock
Prop firm / fund licence	$30,000 – $70,000	Firm-wide MT5 licence; royalty on funded accounts optional
API-as-a-service (signals only)	$100 – $300/month	Supabase neural signals + GMMA Forex computation layer
Ongoing maintenance retainer	$1,000 – $2,500/month	Updates, new pair onboarding, broker compatibility patches

$18K–$35K	$55K–$110K	$200–$500
Standalone Sale Forex module only	Full Bundle All three HVTS systems	SaaS / Month Per subscriber

32

Full System Comparison: All Three HVTS Products

Attribute	Liquidity Protocol (Spot)	Futures Fund Mgr	AI-FOREX
Market Type	Gate.io Spot	Gate.io Perp Futures	MT5 Forex Spot
Signal Protocol	Triple OR	Quadruple AND	Quadruple AND
Signal Sources	Neural + GMMA D1/H4/H1 (OR pairs)	Neural + GMMA 1D/4H/1H (AND)	Neural + GMMA D1/H4/H1 (AND)
Neural Source	Per-symbol SQLite (local)	Supabase PostgreSQL (cloud)	Supabase PostgreSQL (cloud)

Attribute	Liquidity Protocol (Spot)	Futures Fund Mgr	AI-FOREX
Strategies	7 concurrent	1 (Long-DCA)	1 (Long-DCA)
Leverage	None (spot)	1×–100× (default 10×)	1:1–1:500 (default 1:100)
Symbol Count	11+ crypto spot	90+ perp futures	28 Forex pairs
Position Sizing	ATH zone % allocation	Margin % of balance	ATR-normalised lot sizing
Session Filter	No (24/7 crypto)	No (24/7 crypto)	Yes — London/NY/Asia windows
Spread Guard	No	No	Yes — pip threshold filter
Multi-User	Yes — full isolation	No	No (single broker account)
Codebase Size	19,994+ lines (19 modules)	~1,229 lines (1 module)	3,840+ lines (6 modules)
Deployment	Full Python environment	Single executable (PyInstaller)	Python + MT5 terminal

Forex-Specific Risk Considerations

- **Leverage Risk:** Forex leverage amplifies both gains and losses. Default 1:100 leverage means a 1% adverse price move results in a 100% loss on the margin deployed for that stage.
- **Broker Risk:** Execution quality, spread widening, slippage, and re-quotes are broker-dependent. HVTS-AI-FOREX is validated against standard ECN/STP brokers but cannot guarantee execution quality.
- **Session Gap Risk:** Weekend gaps and major news events can cause price to open significantly beyond stop loss levels. The spread guard mitigates but does not eliminate this.
- **Correlation Risk:** EUR and GBP crosses are highly correlated. Simultaneous adverse moves across multiple pairs can cause concurrent DCA stage losses.
- **Regulatory Risk:** Forex leverage limits vary by jurisdiction (e.g. 1:30 ESMA cap in EU). The buyer is solely responsible for compliance with applicable regulations.
- **MT5 Dependency:** Requires a live MT5 terminal connection, reliable internet, Python 3.8+ environment, and an active broker account with API/algorithmic trading enabled.