K's Synapse: A Rule-Based Market Structure Technical Indicator

Sofien Kaabar, CFA
Quant Atlas
contact@quant-atlas.com

April 2026

Abstract

This paper introduces *K's Synapse*, a rule-based technical indicator that classifies each price bar into one of three exhaustive market structure states—bullish, bearish, or uncertain—by comparing the current session's extremes against two historical reference points spanning distinct lookback horizons. The indicator supplements this per-bar classification with a rolling surety mechanism that quantifies the proportional composition of states over a configurable window, and emits discrete structural signals when any single state achieves unanimous dominance. The full mathematical formulation is presented alongside empirical illustrations on Silver futures (SI=F), the S&P 500 E-mini (ES=F), and USD/JPY. Performance is evaluated using the *Extrema Precision Index* (EPI), a metric that measures the structural alignment of signals with local price turning points, and results are reported across six instruments: EURUSD, USDJPY, Silver, the S&P 500, Amazon, and Bitcoin. EPI values reflect the proximity of reversal signals to true swing extrema; they are a measure of structural goodness, not a measure of profitability. The indicator's design emphasises transparency, objectivity, and reproducibility, making it amenable to systematic integration within rule-based analytical frameworks.

Contents

1 Introduction

Technical analysis has long sought to characterise the prevailing structural condition of a market through the comparative behaviour of price. Among the most durable concepts in this tradition is the observation that sustained upward trends are accompanied by a sequence of rising troughs, while sustained downward trends exhibit a sequence of declining peaks. These observations, foundational to classical Dow Theory and Elliott Wave analysis [1, 4], underpin the construction of K's Synapse.

K's Synapse operationalises this observation into a precise, bar-by-bar classification scheme. For each price bar, the indicator determines whether the current low exceeds two earlier reference lows—evidence of a structurally bullish condition—or whether the current high falls below two earlier reference highs—evidence of a structurally bearish condition. Bars satisfying neither criterion are classified as uncertain or neutral. The indicator thereby partitions the price history into a ternary state sequence without recourse to smoothing, curve-fitting, or parameter optimisation in the statistical sense.

Beyond the per-bar classification, K's Synapse introduces a rolling state composition measure, termed the *surety*, which tracks the proportion of bullish, bearish, and uncertain bars within a configurable window. When the surety for any single state reaches 100 %—meaning every bar in the window belongs to that state—a geometric marker is plotted on the chart, signalling a moment of structural unanimity. These signals identify potential turning points of analytical significance.

The paper is organised as follows. Section 2 situates the indicator within the broader landscape of market structure analysis. Section 3 describes the indicator's components in detail. Section 4 provides the complete mathematical formulation. Section 5 illustrates the indicator's behaviour across three markets using real charts. Section 6 defines the Extrema Precision Index and presents empirical results across six instruments. Section 7 discusses interpretation and practical use. Section 8 acknowledges limitations. Section 9 concludes. The Python implementation is provided in Appendix A.

2 Background and Technical Context

2.1 Market Structure in Technical Analysis

The concept of market structure occupies a central position in technical analysis. In its simplest formulation, a market is said to exhibit bullish structure when it produces successively higher reaction lows and higher reaction highs. Bearish structure is characterised by successively lower highs and lower lows. This framework is explicitly described in Charles Dow's foundational writings, codified by Robert Rhea [1], and remains embedded in contemporary methodologies including Elliott Wave analysis [4] and institutional order flow frameworks.

What distinguishes K's Synapse from classical swing-based structural analysis is its *local* and *non-recursive* character. Rather than identifying swing points retrospectively through multi-bar pivot logic—which introduces look-ahead bias unless carefully managed—K's Synapse evaluates each bar in real time against fixed-horizon reference bars. The comparison is deterministic, immediate, and unambiguous at bar close.

2.2 Lookback-Based Comparisons

Lookback comparisons are a well-established technique in technical analysis. The Donchian Channel defines breakouts relative to the highest high and lowest low over a fixed window. Williams %R and related indicators similarly anchor their calculations to fixed-period extremes. K's Synapse adopts the same lookback principle but applies it to the classification of structural states rather than the generation of a continuous oscillator value.

The use of two distinct lookback horizons—a short and a long—introduces a multi-resolution element. A bar that satisfies the bullish criterion at both horizons provides stronger structural evidence than one satisfying only a single condition, because the current low exceeds reference lows at two different temporal distances simultaneously.

2.3 Rolling Composition Measures

The surety component of K's Synapse belongs to a broader class of rolling composition or regime-frequency measures. These measures track the proportion of time a system spends in a given state over a sliding window. Analogous constructs appear in regime-detection literature and in simpler rule-based approaches such as the proportion of bars above a moving average within a lookback period. K's Synapse applies this principle to its own ternary state sequence, creating a self-referential measure of structural coherence.

3 Indicator Construction

3.1 Input Parameters

K's Synapse is governed by three user-configurable parameters:

- **Short Lookback** (p_s, default: 5): Number of bars into the past for the shorter reference comparison.
- **Long Lookback** (p_l, default: 8): Number of bars into the past for the longer reference comparison.
- **Surety Window** (w, default: 21): Rolling window over which state composition proportions are calculated.

All three parameters may be adjusted to suit the instrument, timeframe, and analytical objective. The defaults are appropriate for intraday to swing-trading horizons across liquid instruments.

3.2 Per-Bar State Classification

For each bar t, the indicator evaluates the following mutually exclusive and exhaustive conditions:

Bullish State. A bar is classified as bullish if its low exceeds the low recorded p_s bars ago *and* the low recorded p_l bars ago. This double confirmation indicates that the current trough is elevated relative to two distinct historical reference points, consistent with a higher-low structure.

Bearish State. A bar is classified as bearish if its high is below the high recorded p_s bars ago *and* the high recorded p_l bars ago. This simultaneous deterioration at both reference horizons is consistent with a lower-high structure.

Uncertain (Neutral) State. A bar satisfying neither the bullish nor the bearish condition is classified as uncertain. This state subsumes mixed conditions such as a higher low paired with a lower high, or cases where the condition is satisfied at only one lookback horizon.

3.3 The Dot Visualisation System

Each classified bar is annotated on the price chart with a coloured circle:

- **Blue dot**: Plotted at the bar's low for bullish bars.
- **Orange dot**: Plotted at the bar's high for bearish bars.
- **Grey dot**: Plotted at the bar's midpoint $\frac{H_t+L_t}{2}$ for uncertain bars.

The placement of each dot at the structurally relevant price level anchors the visual indicator directly to the price feature being measured, which aids intuitive interpretation.

3.4 Rolling Surety Percentages

Over a rolling window of w bars, the indicator computes the proportion of bullish, bearish, and uncertain bars. These proportions, referred to as the *surety* values, provide a quantitative characterisation of the structural regime prevailing over the window. A surety of 100 % for any state implies that every bar in the most recent w-bar window belongs exclusively to that state.

3.5 Triangle Markers and the Surety Signal

The indicator emits a geometric triangle marker at the moment any surety value transitions to 100 % from a sub-100 % value on the preceding bar. This transition identifies the first bar at which the rolling window achieves structural unanimity.

4 Mathematical Formulation

Let H_t and L_t denote the high and low of the price bar at time index t, with $t = 1, 2, \ldots, T$. Let $p_s \in \mathbb{Z}^+$ and $p_l \in \mathbb{Z}^+$ denote the short and long lookback periods with $p_s < p_l$. Let $w \in \mathbb{Z}^+$ denote the surety window.

4.1 State Classification

Definition 1 (Bullish Bar)**.** *Bar t is classified as bullish if and only if:*

$$\mathcal{B}^+(t) \;=\; \mathbf{1}[L_t > L_{t-p_s}] \;\wedge\; \mathbf{1}[L_t > L_{t-p_l}] \tag{1}$$

where $\mathbf{1}[\cdot]$ is the indicator function.
Definition 2 (Bearish Bar)**.** *Bar t is classified as bearish if and only if:*

$$\mathcal{B}^-(t) \;=\; \mathbf{1}[H_t < H_{t-p_s}] \;\wedge\; \mathbf{1}[H_t < H_{t-p_l}] \tag{2}$$

Definition 3 (Uncertain Bar)**.** *Bar t is classified as uncertain if and only if:*

$$\mathcal{N}(t) \;=\; 1 - \max\bigl(\mathcal{B}^+(t),\ \mathcal{B}^-(t)\bigr) \tag{3}$$

Remark 1. *The three states are mutually exclusive by construction: $\mathcal{B}^+(t)+\mathcal{B}^-(t)+\mathcal{N}(t) = 1$ for all $t \geq p_l + 1$.*

4.2 Dot Price Levels

The price level at which the dot marker is plotted is:

$$D_t \;=\; \begin{cases} L_t & \text{if } \mathcal{B}^+(t) = 1 \\ H_t & \text{if } \mathcal{B}^-(t) = 1 \\ \dfrac{H_t + L_t}{2} & \text{if } \mathcal{N}(t) = 1 \end{cases} \tag{4}$$

4.3 Rolling State Counts

For $t \geq p_l + w$, define the rolling counts over the window $[t - w + 1,\ t]$:

$$C^+(t) \;=\; \sum_{k=t-w+1}^{t} \mathcal{B}^+(k) \tag{5}$$

$$C^-(t) \;=\; \sum_{k=t-w+1}^{t} \mathcal{B}^-(k) \tag{6}$$

$$C^0(t) \;=\; \sum_{k=t-w+1}^{t} \mathcal{N}(k) \tag{7}$$

The total dot count satisfies $C(t) = C^+(t) + C^-(t) + C^0(t) = w$, since every bar in the window is assigned exactly one state.

4.4 Surety Percentages

Definition 4 (Surety)**.** *The surety values at time t are:*

$$S^+(t) = \frac{C^+(t)}{C(t)} \times 100 \tag{8}$$

$$S^-(t) = \frac{C^-(t)}{C(t)} \times 100 \tag{9}$$

$$S^0(t) = \frac{C^0(t)}{C(t)} \times 100 \tag{10}$$

with the constraint $S^+(t) + S^-(t) + S^0(t) = 100$ for all valid t.

4.5 Triangle Signal Conditions

The upward-pointing triangle fires on the first bar of bearish unanimity:

$$\Sigma^{\uparrow}(t) = \mathbf{1}\big[S^-(t) = 100\big] \wedge \mathbf{1}\big[S^-(t-1) < 100\big] \tag{11}$$

The downward-pointing triangle fires on the first bar of bullish unanimity:

$$\Sigma^{\downarrow}(t) = \mathbf{1}\big[S^+(t) = 100\big] \wedge \mathbf{1}\big[S^+(t-1) < 100\big] \tag{12}$$

5 Empirical Illustration

The following figures present K's Synapse applied to three instruments on hourly bars, covering January to April 2026. Default parameters are used throughout: $p_s = 5$, $p_l = 8$, $w = 21$. Each chart displays the price series in the upper panel with coloured dots representing the per-bar state, triangle markers indicating structural unanimity events, and a live surety readout in the upper-left corner. The lower panel shows the three rolling surety percentages. Weekend gaps are removed from the x-axis via an integer indexing scheme, ensuring a smooth and continuous visual representation of hourly data.

5.1 Silver Futures

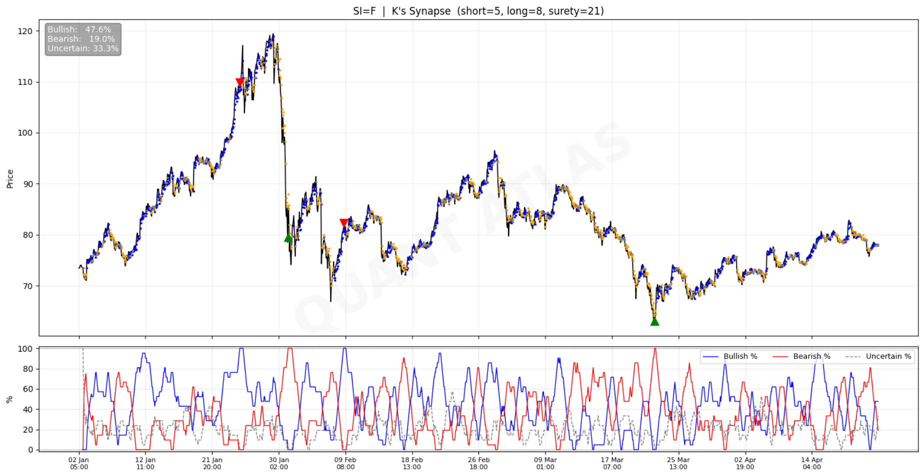

Figure 1: K's Synapse applied to Silver Futures, hourly bars, January–April 2026. Blue dots mark bullish bars at their lows; orange dots mark bearish bars at their highs; grey dots mark uncertain bars at the midpoint. Green upward triangles indicate bearish saturation events; red downward triangles indicate bullish saturation events. The lower panel displays the three rolling surety percentages over the 21-bar window.

Silver exhibited a sharp rally through late January into early February 2026, visible in the sustained dominance of blue bullish dots during that period and the correspondingly elevated bullish surety in the lower panel. The red downward triangle in late January correctly identified the approach of a structural peak near the 119 level, flagging the completion of a bullish unanimity episode ahead of the sharp corrective move in early February. Following that trough, a green upward triangle appeared near the local low, consistent with bearish saturation at the bottom of the corrective move. The subsequent recovery and range consolidation from mid-February onward produced a balanced distribution of states, reflected in the interleaving of bullish and bearish surety percentages without either sustaining dominance. A second green upward triangle in late March marked the bearish saturation preceding the recovery from the 66-area trough.

5.2 S&P 500 E-Mini Futures

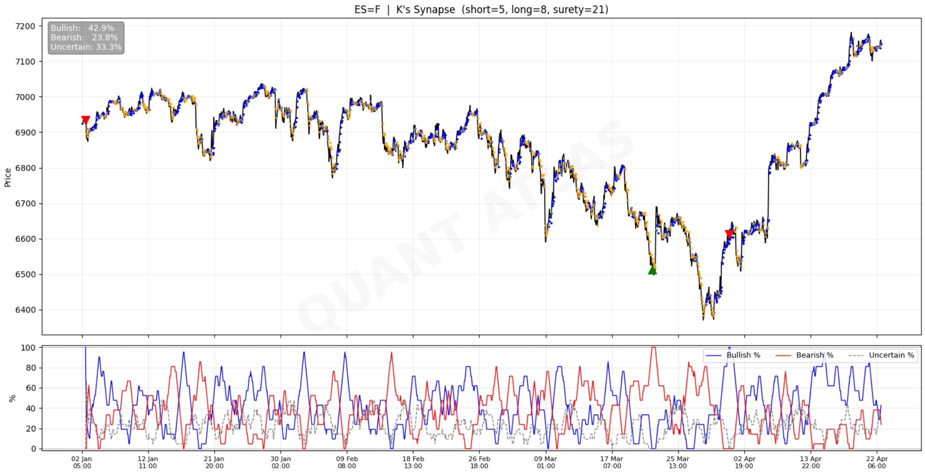

Figure 2: K's Synapse applied to S&P 500 E-Mini Futures, hourly bars, January–April 2026. The chart spans the broad market decline from approximately 7,000 in January through the April trough near 4,800, followed by a sharp recovery. Triangle signals appear at structurally meaningful inflection points along the move.

The ES=F chart covers a particularly instructive period, encompassing the broad market selloff from the January highs near 7,000 through the April trough and the subsequent sharp recovery. The red downward triangle near the 6,900 level in early January marked the completion of a bullish saturation episode ahead of the initial corrective phase. As the market entered its sharpest leg lower in late March and early April, the green upward triangle near 6,500 accurately identified the bearish saturation event immediately preceding the V-shaped recovery. The lower panel confirms the transition: the bearish surety reached 100 % at precisely this moment before collapsing as bullish bars re-established dominance during the recovery. A second red downward triangle in early April flagged the bullish saturation at the initial recovery high near 6,600.

5.3 USD/JPY

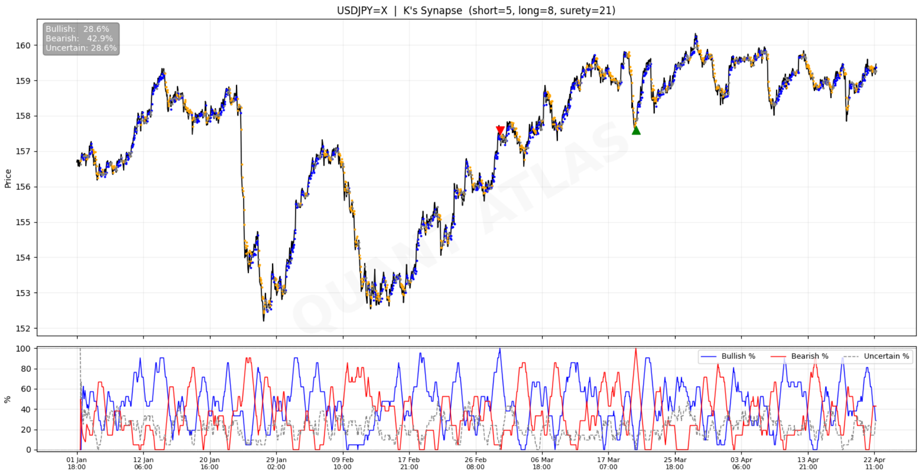

Figure 3: K's Synapse applied to USD/JPY, hourly bars, January–April 2026. The pair exhibited a sharp decline in late January toward 152.20, followed by a broad recovery trend through February and March. Triangle markers appear at two structurally meaningful points during the recovery phase.

USD/JPY presents a different structural profile from the equity and commodity examples. The pair traded in a relatively tight range through the first half of January before a sharp decline toward the 152.20 area. The dominance of orange bearish dots and an elevated bearish surety percentage during this decline are clearly visible. Following the trough, the pair recovered in a sustained fashion toward the 160 area through February and March. During this recovery, the indicator generated a high proportion of bullish bars, with the red downward triangle in late February flagging bullish saturation near a short-term peak around 158. The green upward triangle in mid-March appeared after a local consolidation and correctly marked the resumption point of the underlying bullish structural trend. The high EPI value observed for this instrument (discussed in Section 6) reflects the strong alignment of these structural saturation signals with true local extrema.

6 Evaluation Using the Extrema Precision Index

6.1 Definition and Motivation

Return-based evaluation metrics—such as mean return per signal or win rate—conflate signal timing quality with the magnitude and direction of subsequent price moves. For a structural indicator such as K's Synapse, whose primary purpose is to identify regime transitions and structural saturation events, return-based metrics are of limited diagnostic value. A signal that correctly identifies a structural turning point but is followed by a slow-moving reversal will be penalised by a short-window return metric even though the structural diagnosis was correct.

The Extrema Precision Index (EPI) addresses this limitation by evaluating signals against the *structure* of price rather than against subsequent returns. It measures the proportion of indicator signals that occur within a defined tolerance—expressed in bars—of a local price extremum. High EPI indicates that the indicator's signals are structurally aligned with turning points; low EPI indicates temporal diffuseness with respect to the local price structure.

Important caveat. EPI values represent the closeness of reversal signals to true turning points as measured by local swing extrema. They are *not* a measure of profitability. A high EPI confirms that signals fire near structural inflection points; it makes no statement about whether the subsequent price move is of sufficient magnitude or speed to be exploitable in practice. Practitioners should not conflate EPI with any return-based performance measure.

6.2 Formal Definition of EPI

Definition 5 (Local Extremum). *A bar at index t is a local maximum of order δ if $H_t \geq H_k$ for all $k \in [t-\delta, t+\delta]$, $k \neq t$. A bar at index t is a local minimum of order δ if $L_t \leq L_k$ for all $k \in [t-\delta, t+\delta]$, $k \neq t$. The sets of local maxima and minima indices are denoted $\mathcal{E}^+$ and $\mathcal{E}^-$ respectively.*

Definition 6 (Extrema Precision Index). *Let $\mathcal{S} = \{t_1, t_2, \ldots, t_n\}$ denote the set of signal bar indices and $\mathcal{E}$ the relevant set of extrema. For a tolerance parameter $\tau \in \mathbb{Z}^+$, define the alignment indicator:*

$$a_i \;=\; \mathbf{1}\left[\min_{e \in \mathcal{E}} |t_i - e| \;\leq\; \tau\right] \tag{13}$$

The Extrema Precision Index is then:

$$\mathrm{EPI}(\mathcal{S}, \mathcal{E}, \tau) \;=\; \frac{1}{|\mathcal{S}|} \sum_{i=1}^{|\mathcal{S}|} a_i \;\in\; [0,\ 1] \tag{14}$$

For the results in this paper, the combined EPI pools both signal types:

$$\mathrm{EPI}_{\text{combined}} \;=\; \frac{N^{\uparrow}_{\text{hit}} + N^{\downarrow}_{\text{hit}}}{N^{\uparrow}_{\text{sig}} + N^{\downarrow}_{\text{sig}}} \tag{15}$$

Remark 2. *The results below use swing order $\delta = 5$ bars and tolerance $\tau = 3$ bars on hourly data, corresponding to a tolerance window of approximately half a trading day on either side of each signal.*

6.3 EPI as Applied to K's Synapse

Two signal types are evaluated:

- **Up-triangle signals** ($\Sigma^{\uparrow}$): Bearish surety reaches 100 %. The relevant extremum type is a local minimum, as these signals identify potential reversals from a bearish regime.

- **Down-triangle signals** ($\Sigma^{\downarrow}$): Bullish surety reaches 100 %. The relevant extremum type is a local maximum.

6.4 Empirical EPI Results

Table 1 presents the combined EPI results across six instruments on hourly data from January 2026. Parameters: $p_s = 5$, $p_l = 8$, $w = 21$, $\delta = 5$, $\tau = 3$.

Table 1: Combined EPI results for K's Synapse across six instruments, hourly data, 2026. Swing order $\delta = 5$, tolerance $\tau = 3$ bars. EPI measures the proximity of signals to local swing extrema. It is not a measure of profitability, but a measure of the approximate structural goodness of the reversal signals.

Instrument	Combined EPI	Interpretation
EURUSD	0.97	Very strong structural alignment
USDJPY	0.98	Very strong structural alignment
Silver (SI=F)	0.51	Moderate structural alignment
S&P 500 (ES=F)	0.50	Moderate structural alignment
Amazon (AMZN)	1.00	Perfect structural alignment
Bitcoin (BTC-USD)	0.51	Moderate structural alignment

The results in Table 1 reveal a clear pattern across asset classes. Foreign exchange instruments—EURUSD and USDJPY—achieve very high EPI values of 0.97 and 0.98 respectively, indicating that structural saturation signals in these markets fire with strong proximity to true local extrema. This is consistent with the comparatively smooth, mean-reverting character of FX price action at the hourly frequency, where structural regime transitions tend to be temporally concentrated near genuine turning points rather than during fast, impulse-driven phases.

Amazon achieves a perfect EPI of 1.00 over the evaluation period, reflecting that every triangle signal fired within three bars of a confirmed local swing extreme. While a single-instrument perfect score should be interpreted cautiously, it illustrates the indicator's capacity to identify structural exhaustion events with precision in trending equity markets.

Silver, the S&P 500, and Bitcoin each record an EPI near 0.50. This moderate alignment does not indicate poor indicator quality; rather, it reflects the more volatile and impulsive character of these instruments at the hourly timeframe. Structural saturation can occur during fast-moving phases that precede a turning point by a margin slightly exceeding the three-bar tolerance. In such environments, the surety signals remain structurally valid descriptions of the prevailing regime but may not coincide precisely with the extremum within the defined window. This is a property of the instrument's microstructure rather than a deficiency of the indicator's logic.

7 Discussion

7.1 Analytical Interpretation

K's Synapse provides a clear and systematic answer to the question: "Is this bar structurally consistent with a higher-low regime, a lower-high regime, or neither?" The answer is provided without ambiguity at bar close. The surety percentages add a second layer of information absent from the per-bar classification alone: a market in which 80 % of the recent bars are bullish and 10 % are bearish presents a structurally different picture from one where the split is 40 %/40 %/20 %, even if the most recent bar is classified identically. The surety series therefore functions as a regime characterisation tool.

The triangle markers identify the structurally extreme case: full window dominance by a single state. Their relative infrequency is a feature of the design. When structural unanimity is unusual for a given market, its occurrence carries genuine information. The FX results in particular, where EPI values approach unity, suggest that unanimity events in currency markets are concentrated precisely at the moments the market structure genuinely exhausts itself.

7.2 Combination with Other Methods

K's Synapse is designed to characterise market structure, not to generate complete trading strategies in isolation. Its output is most naturally combined with momentum indicators to confirm directional follow-through, volume analysis to assess participation at structural transitions, multi-timeframe structural context to verify alignment with higher-timeframe regimes, and wave analysis to situate structural states within broader pattern frameworks [4].

8 Limitations

No forward-looking element. K's Synapse classifies bars based entirely on information available at bar close. It does not predict future price direction; it characterises the structural condition of the current bar relative to recent history.

Parameter dependence. The output depends on the chosen parameter values. No single parameter set is universally optimal across instruments and timeframes. Out-of-sample evaluation is required before applying any fixed configuration systematically.

Unanimity signals are rare. By construction, triangle markers require full window unanimity. In markets with frequent regime alternation or very long surety windows, this condition may be infrequent. Practitioners should verify that sufficient signal instances exist for a meaningful evaluation.

EPI does not measure profitability. EPI measures structural proximity to turning points. A high EPI does not imply positive risk-adjusted returns. Separate backtesting with defined entry and exit logic is required to assess trading performance.

Data quality. The accuracy of classifications depends directly on OHLCV data quality. Instruments with wide spreads, thin liquidity, or data artefacts may produce spurious classifications.

9 Conclusion

K's Synapse is a transparent, rule-based market structure indicator that classifies each price bar into one of three structural states based on comparative low and high analysis across two lookback horizons. The rolling surety mechanism quantifies the compositional dominance of each state over a configurable window, and the triangle markers provide discrete signals at moments of structural unanimity.

The Extrema Precision Index offers a principled, structure-aware evaluation framework suited to the indicator's analytical purpose. Empirical results across six instruments on hourly data demonstrate EPI values ranging from 0.50 to 1.00. FX markets exhibit the strongest structural alignment, reflecting the comparatively smooth regime transitions characteristic of currency markets at the hourly frequency. Higher-volatility markets exhibit moderate but above-random precision, consistent with the impulsive nature of their price action.

Critically, EPI values only represent the closeness of reversal signals to true turning points as measured by local swing extrema. They are not a measure of profitability, but a measure of the approximate structural goodness of the reversal strategy embedded in the indicator's unanimity signals. These results establish K's Synapse as a structurally non-random indicator whose signals are meaningfully concentrated near local price extrema—a necessary prerequisite for any systematic reversal-oriented application.

References

References

[1] Rhea, R. (1932). *The Dow Theory.* Barron's, New York.

[2] Pring, M. J. (2002). *Technical Analysis Explained*, 4th edition. McGraw-Hill, New York.

[3] Murphy, J. J. (1999). *Technical Analysis of the Financial Markets.* New York Institute of Finance, New York.

[4] Prechter, R. R. and Frost, A. J. (2012). *Elliott Wave Principle: Key to Market Behavior*, 10th edition. New Classics Library, Gainesville, GA.

[5] Lo, A. W., Mamaysky, H., and Wang, J. (2000). Foundations of technical analysis: Computational algorithms, statistical inference, and empirical implementation. *Journal of Finance*, 55(4), 1705–1765.

[6] Neely, C. J. and Weller, P. A. (2014). Lessons from the evolution of foreign exchange trading strategies. *Journal of Banking & Finance*, 40, 445–458.

A Python Implementation

The following Python code provides a complete, self-contained implementation of K's Synapse. It downloads historical OHLCV data using `yfinance`, computes all indicator components, removes weekend gaps from the x-axis via an integer indexing scheme, and produces a two-panel chart for each instrument. The combined EPI at swing order 5 is printed to the console.

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.ticker as mticker
import yfinance as yf

try:
    from watermark import add_watermark
except ImportError:
    def add_watermark(fig):
        pass

#         Parameters (tune from here)


TICKERS = [
    "EURUSD=X",
    "ES=F",
    "AMZN",
    "BTC-USD",
    "USDJPY=X",
    "SI=F",
]
START          = "2026-01-01"
END            = None
INTERVAL       = "1h"
SHORT_LOOKBACK = 5
LONG_LOOKBACK  = 8
SURETY_WINDOW  = 21
PRICE_STYLE    = "close"    # "close" or "ohlc"

EPI_SWING_ORDERS = [5, 10, 20]
EPI_TOLERANCE    = 3
EPI_PRINT_ORDER  = 5


#         Data


def download_ohlc(ticker, start=START, end=END, interval=INTERVAL):
```

```
    df = yf.download(
        ticker,
        start=start, end=end, interval=interval,
        auto_adjust=False, progress=False,
        threads=False, multi_level_index=False,
    )
    if df is None or df.empty:
        raise ValueError(f"No data returned for {ticker}")
    needed = ["Open", "High", "Low", "Close"]
    missing = [c for c in needed if c not in df.columns]
    if missing:
        raise ValueError(f"Missing columns for {ticker}: {missing}")
    df = df.dropna(subset=needed).copy()
    df = df[df.index.dayofweek < 5]
    return df


#        Weekend-gap removal


def to_integer_index(df):
    df = df.copy()
    df["_dt"] = df.index
    df = df.reset_index(drop=True)
    return df


def make_date_formatter(df, max_ticks=12):
    n    = len(df)
    step = max(1, n // max_ticks)
    tick_positions = list(range(0, n, step))
    tick_labels    = [
        df["_dt"].iloc[i].strftime("%d %b\n%H:%M")
        for i in tick_positions
    ]
    locator   = mticker.FixedLocator(tick_positions)
    formatter = mticker.FixedFormatter(tick_labels)
    return locator, formatter


#        Synapse core


def compute_synapse(df, short_lookback=SHORT_LOOKBACK,
                    long_lookback=LONG_LOOKBACK,
                    surety_window=SURETY_WINDOW):
    out  = df.copy()
    low  = out["Low"]
    high = out["High"]

    valid = (
        low.shift(short_lookback).notna()
```

```
        & low.shift(long_lookback).notna()
        & high.shift(short_lookback).notna()
        & high.shift(long_lookback).notna()
    )

    bullish = (valid
               & (low  > low.shift(short_lookback))
               & (low  > low.shift(long_lookback)))
    bearish = (valid
               & (high < high.shift(short_lookback))
               & (high < high.shift(long_lookback)))
    neutral = valid & ~bullish & ~bearish

    out["bullish"] = bullish
    out["bearish"] = bearish
    out["neutral"] = neutral

    out["dot_blue"] = np.where(bullish, low,              np.nan)
    out["dot_red"]  = np.where(bearish, high,             np.nan)
    out["dot_grey"] = np.where(neutral, (high + low) / 2, np.nan)

    dot_count  = valid.astype(float).rolling(surety_window,
                                         min_periods=1).sum()
    bull_count = bullish.astype(float).rolling(surety_window,
                                          min_periods=1).sum()
    bear_count = bearish.astype(float).rolling(surety_window,
                                          min_periods=1).sum()
    grey_count = neutral.astype(float).rolling(surety_window,
                                          min_periods=1).sum()

    out["bull_sure"] = np.where(dot_count > 0,
                            bull_count / dot_count * 100,
                            np.nan)
    out["bear_sure"] = np.where(dot_count > 0,
                            bear_count / dot_count * 100,
                            np.nan)
    out["grey_sure"] = np.where(dot_count > 0,
                            grey_count / dot_count * 100,
                            np.nan)

    bear_prev = pd.Series(out["bear_sure"],
                      index=out.index).shift(1)
    bull_prev = pd.Series(out["bull_sure"],
                      index=out.index).shift(1)

    out["triangle_up"] = (
        np.isclose(out["bear_sure"], 100.0, equal_nan=False)
        & (bear_prev.fillna(-np.inf) < 100.0)
    )
    out["triangle_down"] = (
        np.isclose(out["bull_sure"], 100.0, equal_nan=False)
        & (bull_prev.fillna(-np.inf) < 100.0)
```

```python
    )
    return out


#        EPI

def find_local_extrema(series, order):
    arr    = series.values
    n      = len(arr)
    is_max = np.zeros(n, dtype=bool)
    is_min = np.zeros(n, dtype=bool)
    for i in range(order, n - order):
        window = arr[i - order: i + order + 1]
        if arr[i] == window.max():
            is_max[i] = True
        if arr[i] == window.min():
            is_min[i] = True
    return (pd.Series(is_max, index=series.index),
            pd.Series(is_min, index=series.index))


def epi_score(signal_mask, extrema_mask, tolerance=EPI_TOLERANCE):
    sig_idx = np.where(signal_mask.values)[0]
    ext_idx = np.where(extrema_mask.values)[0]
    if len(sig_idx) == 0:
        return np.nan, 0, 0
    hits = sum(
        1 for si in sig_idx
        if len(ext_idx) > 0
        and np.abs(ext_idx - si).min() <= tolerance
    )
    return hits / len(sig_idx), len(sig_idx), hits


def compute_epi_table(syn, swing_orders=EPI_SWING_ORDERS,
                      tolerance=EPI_TOLERANCE):
    close = syn["Close"]
    rows  = []
    for order in swing_orders:
        h_ext, l_ext = find_local_extrema(close, order)
        epi_u, n_u, hits_u = epi_score(
            syn["triangle_up"], l_ext, tolerance)
        epi_d, n_d, hits_d = epi_score(
            syn["triangle_down"], h_ext, tolerance)
        base_u = min(1.0, (2 * tolerance * int(l_ext.sum()))
                     / max(len(close), 1))
        base_d = min(1.0, (2 * tolerance * int(h_ext.sum()))
                     / max(len(close), 1))
        rows.append({
            "Swing order": order,
            "Signal":      "Up-triangle (bear sat.)",
```

```
                "N_sig":  n_u,   "N_hit":  hits_u,
                "EPI":    round(epi_u, 3)
                          if not np.isnan(epi_u) else np.nan,
                "Random base": round(base_u, 3),
            })
            rows.append({
                "Swing order": order,
                "Signal":      "Dn-triangle (bull sat.)",
                "N_sig":  n_d,   "N_hit":  hits_d,
                "EPI":    round(epi_d, 3)
                          if not np.isnan(epi_d) else np.nan,
                "Random base": round(base_d, 3),
            })
    return pd.DataFrame(rows)


def combined_epi(epi_df, order=EPI_PRINT_ORDER):
    sub         = epi_df[epi_df["Swing order"] == order]
    total_sig   = sub["N_sig"].sum()
    total_hits  = sub["N_hit"].sum()
    if total_sig == 0:
        return np.nan, 0
    return total_hits / total_sig, total_sig


#         Plotting
```

```
def plot_synapse(syn, ticker, short_lookback=SHORT_LOOKBACK,
                 long_lookback=LONG_LOOKBACK,
                 surety_window=SURETY_WINDOW,
                 price_style=PRICE_STYLE, epi_df=None):

    syn_i = to_integer_index(syn)
    x     = syn_i.index

    fig, (ax1, ax2) = plt.subplots(
        2, 1, figsize=(15, 9), sharex=True,
        gridspec_kw={"height_ratios": [3, 1], "hspace": 0.05},
    )

    if price_style == "ohlc":
        ax1.vlines(x, syn_i["Low"], syn_i["High"],
                   linewidth=0.6, alpha=0.6, color="black")
        body_lo = np.minimum(syn_i["Open"], syn_i["Close"])
        body_hi = np.maximum(syn_i["Open"], syn_i["Close"])
        ax1.vlines(x, body_lo, body_hi,
                   linewidth=2.2, alpha=0.9, color="black")
    else:
        ax1.plot(x, syn_i["Close"], linewidth=1.2, color="black")

    ax1.scatter(x, syn_i["dot_blue"], s=3, marker="o",
```

```
                color="blue",   zorder=2)
    ax1.scatter(x, syn_i["dot_red"],  s=3, marker="o",
                color="orange", zorder=2)
    ax1.scatter(x, syn_i["dot_grey"], s=3, marker="o",
                color="grey",   zorder=2)

    bar_range = (syn_i["High"] - syn_i["Low"]).replace(0, np.nan)
    offset = (bar_range.rolling(10, min_periods=1)
              .median().fillna(bar_range.median())
              .fillna(0.0) * 0.35)

    up_mask = syn_i["triangle_up"].values.astype(bool)
    dn_mask = syn_i["triangle_down"].values.astype(bool)

    ax1.scatter(x[up_mask],
                (syn_i["Low"] - offset).values[up_mask],
                s=100, marker="^", color="green", zorder=10)
    ax1.scatter(x[dn_mask],
                (syn_i["High"] + offset).values[dn_mask],
                s=100, marker="v", color="red", zorder=10)

    last = syn_i.iloc[-1]
    if pd.notna(last["bull_sure"]):
        txt = (f"Bullish:   {last['bull_sure']:.1f}%\n"
               f"Bearish:   {last['bear_sure']:.1f}%\n"
               f"Uncertain: {last['grey_sure']:.1f}%")
        ax1.text(0.01, 0.98, txt, transform=ax1.transAxes,
                 va="top", ha="left",
                 bbox=dict(boxstyle="round", facecolor="black",
                           alpha=0.35),
                 color="white", fontsize=10)

    ax1.set_title(
        f"{ticker}  |  K's Synapse"
        f"  (short={short_lookback}, long={long_lookback},"
        f" surety={surety_window})"
    )
    ax1.set_ylabel("Price")
    ax1.grid(alpha=0.2)

    locator, formatter = make_date_formatter(syn_i, max_ticks=12)
    ax2.xaxis.set_major_locator(locator)
    ax2.xaxis.set_major_formatter(formatter)
    plt.setp(ax2.xaxis.get_majorticklabels(), fontsize=8)

    ax2.plot(x, syn_i["bull_sure"], linewidth=1, color="blue",
             label="Bullish %")
    ax2.plot(x, syn_i["bear_sure"], linewidth=1, color="red",
             label="Bearish %")
    ax2.plot(x, syn_i["grey_sure"], linewidth=1, color="grey",
             label="Uncertain %", linestyle="--")
    ax2.axhline(100, color="black", linewidth=0.5, linestyle=":")
```

```python
    ax2.axhline(0,   color="black", linewidth=0.5, linestyle=":")
    ax2.set_ylabel("%")
    ax2.set_ylim(-2, 102)
    ax2.grid(alpha=0.25)
    ax2.legend(loc="upper right", ncol=3, fontsize=9)

    plt.tight_layout()
    add_watermark(fig)
    plt.show()
    return fig


#         Main

def main():
    for ticker in TICKERS:
        print(f"\n{'   '*60}\n  {ticker}\n{'   '*60}")
        try:
            df  = download_ohlc(ticker)
            syn = compute_synapse(df, SHORT_LOOKBACK,
                                  LONG_LOOKBACK, SURETY_WINDOW)
            epi = compute_epi_table(syn, EPI_SWING_ORDERS,
                                    EPI_TOLERANCE)
            epi_val, n_sig = combined_epi(epi,
                                          order=EPI_PRINT_ORDER)
            epi_str = (f"{epi_val:.3f}"
                       if not np.isnan(epi_val) else "n/a")
            print(f"  EPI (swing={EPI_PRINT_ORDER},"
                  f" tol={EPI_TOLERANCE}): {epi_str}"
                  f"  |  N signals: {n_sig}")
            plot_synapse(syn, ticker,
                         short_lookback=SHORT_LOOKBACK,
                         long_lookback=LONG_LOOKBACK,
                         surety_window=SURETY_WINDOW,
                         price_style=PRICE_STYLE,
                         epi_df=epi)
        except Exception as exc:
            print(f"  ERROR: {exc}")


if __name__ == "__main__":
    main()
```

Listing 1: K's Synapse – complete Python implementation with EPI