PSEDYS | TECHNICAL REFERENCE MANUAL V3.0

ATR — Average True Range

ATR — Average True Range

Technical Reference Manual v1.0

PSEDYS | Complete Edition
True Range · Wilder RMA · SL Sizing · Absolute Volatility · ATR vs StdDev · MTF · Bot vs TradingView

CONTENTS

1. What is ATR? Concept and Origin
2. Calculation Formula — Complete
   - 2.1 True Range (TR)
   - 2.2 ATR = Wilder RMA(TR, N)
   - 2.3 Complete Numerical Example — 50 bars
3. Range and Interpretation
4. Lag and Warm-up
5. ATR Uses in Trading
   - 5.1 Position Sizing based on ATR
   - 5.2 Dynamic Stop Loss (ATR Stop)
   - 5.3 Trailing Stop
   - 5.4 Target (TP) based on ATR
6. ATR vs StdDev — Comparison
7. ATR in Sideways and Trending Markets
8. Multi-Timeframe Analysis (MTF)
9. PSEDYS — ATR as a Basis for SL/TP
10. Bot vs TradingView Validation
11. Practical Exercises
12. Cheat Sheet — Quick Summary

1. What is ATR? Concept and Origin

ATR (Average True Range) was created by **J. Welles Wilder Jr.** and introduced in the book New Concepts in Technical Trading Systems (1978). ATR measures the **absolute volatility** of an instrument — how much the price moves on average over a period.

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

Philosophy: A simple High-Low does not capture overnight gaps or the gap price relative to the previous bar. True Range extends the range to include these situations.

---

2. Calculation Formula — Complete

2.1 True Range (TR)

```
TR[i] = max(
    High[i] − Low[i],           ← intra-bar range
    |High[i] − Close[i−1]|,    ← gap up from previous close
    |Low[i] − Close[i−1]|      ← gap down from previous close
)
```

First bar: TR[1] = High[1] − Low[1] (no previous Close).

When each component dominates:
- `H-L` : normal bar without gap
- `|H-PC|` : gap up (High > PrevClose significantly)
- `|L-PC|` : gap down (Low < PrevClose significantly)

2.2 ATR = Wilder RMA(TR, N)

```
K = 1 / period   ← Wilder smoothing (NOT K=2/(N+1) standard EMA!)

ATR[period] = SMA(TR[1..period])   ← seed: simple average of the first N TR
ATR[i]      = TR[i] × K + ATR[i−1] × (1−K)   for i > period
```

Wilder Equivalent:

```
ATR[i] = (ATR[i−1] × (period−1) + TR[i]) / period
```

Difference from standard EMA:
- EMA: K = 2/(N+1) — for N=14: K=0.1333
- Wilder: K = 1/N — for N=14: K=0.0714 (slower, smoother)

2.3 Complete Numerical Example — period=14, 50 bars

TR bars 1–14:

```
TR[1..14] = [0.87, 2.01, 0.82, 1.37, 1.86, 1.64, 1.54, 1.87, 1.82, 1.93, 0.90, 0.90, 1.26, 2.02]
ATR seed (bar 15) = SMA(TR[1..14]) = 1.4864
```

Verification bar 16:

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

```
High[16]=101.38, Low[16]=99.98, Close[15]=100.67
TR[16] = max(101.38−99.98, |101.38−100.67|, |99.98−100.67|)
       = max(1.400, 0.710, 0.690) = 1.4000

ATR[16] = 1.4000/14 + 1.4864×(13/14)
        = 0.1000 + 1.3803 = 1.4803  ✓
```

ATR Table bars 15–50:

Bar	TR	ATR	Interpretation
15	2.020	**1.4864**	Seed SMA
20	1.480	1.4893	Stable volatility
25	1.370	1.4428	Slightly decreasing
30	1.580	1.5475	Recovery
35	1.970	1.5828	Bullish trend, vol increasing
40	1.170	1.5543	Volatility pullback
45	2.040	1.5426	Volatility spike
50	1.830	1.5829	Stabilization

---

3. Range and Interpretation

ATR is expressed in **absolute price units** (not percentages):
- ATR(BTC) = 1500 USD → average bar moves 1500 USD
- ATR(ETH) = 80 USD → average bar moves 80 USD

Cross-instrument comparisons: use ATR% = ATR/Close × 100

```
Increasing ATR: expanding volatility (active trend or spike)
Decreasing ATR: contracting volatility (consolidation, pre-squeeze)
Stable ATR: market in equilibrium
```

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

4. Lag and Warm-up

```
First ATR value: bar period+1 = bar 15 (seed SMA at bar 14)
Warm-up: period+1 bars = 15 bars for period=14
Wilder Lag: ~period bars (K=1/N much slower than EMA)
  → ATR responds slowly to spikes → advantage: stability
  → Disadvantage: in rapid volatility explosion, ATR initially underestimates
```

5. ATR Uses in Trading

5.1 Position Sizing based on ATR

```
Risk per trade = 1% of capital
ATR_stop_distance = 2 × ATR (distance SL from entry)
Position size = (Capital × Risk%) / (ATR_stop_distance)

Example:
  Capital = 10,000 USD
  ATR(BTC,4H) = 500 USD
  SL = 2 × 500 = 1000 USD below entry
  Position size = (10,000 × 1%) / 1000 = 0.1 BTC
```

5.2 Dynamic Stop Loss (ATR Stop)

```
Long:  SL = Entry − mult × ATR   (standard mult = 1.5–2.0)
Short: SL = Entry + mult × ATR

Advantage vs fixed SL: SL adapts to instrument volatility
  → Volatile market (high ATR): wider SL, noise doesn't take you out
  → Quiet market (low ATR): tighter SL, lower risk
```

5.3 Trailing Stop

```
Long Trailing:
  Trail_SL[i] = max(Trail_SL[i-1], Close[i] − mult × ATR[i])
  → SL rises with price, never lowers

Short Trailing:
  Trail_SL[i] = min(Trail_SL[i-1], Close[i] + mult × ATR[i])
```

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

5.4 Target (TP) based on ATR

```
TP1 = Entry + 1.5 × ATR
TP2 = Entry + 3.0 × ATR
TP3 = Entry + 5.0 × ATR  (or trailing)

Risk:Reward = (TP − Entry) / (Entry − SL)
  With SL=2×ATR, TP1=1.5×ATR: R:R = 0.75 (below 1 → skip)
  With SL=1.5×ATR, TP1=3×ATR: R:R = 2.0 (ok)
```

6. ATR vs StdDev — Comparison

Aspect	ATR	StdDev
Basis	True Range (H/L/PC)	Distance of Close from average
Includes gaps	Yes (	H-PC
Smoothing	Wilder RMA (K=1/N)	None (calculated directly)
Units	Absolute (USD, BTC)	Absolute (relative to SMA)
Usage	SL/TP sizing, volatility	BB, statistical position
Reaction to spike	Gradual (slow Wilder)	Immediate (calculated directly)

7. ATR in Sideways and Trending Markets

Sideways: Low and stable ATR. Small SLs → larger position sizing.

Active trend: ATR gradually increases. SLs must automatically widen if using fixed mult. ATR-based system adapts automatically.

Breakout: ATR spikes — if using pre-breakout fixed SL, risk being taken out by spike. ATR trailing automatically expands.

8. MTF

Level	Timeframe	ATR Usage
Sizing	4H	ATR(4H) for final SL/TP and position sizing
Context	1H	ATR(1H) for intermediate TP1/TP2
Entry	5M	ATR(5M) for entry precision SL (tight)

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

MTF Rule: SL calculated on entry TF (5M) but checked not to be below ATR(4H) × factor. An SL of 0.5×ATR(4H) is too tight to survive noise.

---

9. PSEDYS — ATR as a Basis for SL/TP

In PSEDYS, ATR underpins all SL and trailing calculations:

```
# Dynamic SL
sl_long  = entry_price - atr_mult * atr
sl_short = entry_price + atr_mult * atr

# Trailing update
if direction == 'long':
    new_trail = close - trail_mult * atr
    trailing_sl = max(trailing_sl, new_trail)

# Volatility filter
high_volatility = atr > atr_avg * 1.5  # ATR 50% above average
low_volatility  = atr < atr_avg * 0.7  # ATR 30% below average
```

ATR% for screener:

```
atr_pct = atr / close * 100
# Filtering: avoid symbols with ATR% < 0.5% (too little movement)
# or ATR% > 10% (too volatile for normal sizing)
```

---

10. Bot vs TradingView Validation

```
□ TR = max(H-L, |H-PC|, |L-PC|)  — 3 components
□ Seed: ATR[period] = SMA(TR[1..period]) — simple average
□ Recursive: ATR[i] = TR[i]/N + ATR[i-1]×(N-1)/N  ← Wilder K=1/N
□ NOT K=2/(N+1) standard EMA → cause #1 of divergence
□ First value: bar period+1 (index period, 0-indexed)
□ Tolerance: < 0.001
```

Python Script:

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

```
def atr(highs, lows, closes, period=14):
    n = len(closes)
    tr = [None] * n
    for i in range(1, n):
        tr[i] = max(highs[i]-lows[i],
                    abs(highs[i]-closes[i-1]),
                    abs(lows[i]-closes[i-1]))
    atr_v = [None] * n
    atr_v[period] = sum(tr[1:period+1]) / period  # SMA seed
    for i in range(period+1, n):
        atr_v[i] = tr[i]/period + atr_v[i-1]*(period-1)/period
    return atr_v
```

---

11. Practical Exercises

Exercise 1: SL with ATR in High Volatility Market

Situation: BTC/USDT 4H. ATR(14) = 1800 USD. Long Entry at 65000. Rule: SL = 2×ATR.

Answer: SL = 65000 − 2×1800 = 65000 − 3600 = **61400 USD**. With capital 10000 USD and risk 1%: position size = (10000×0.01)/3600 = **0.0278 BTC**. If ATR rises to 2500, SL automatically widens to 5000 USD → position size decreases to 0.02 BTC.

Exercise 2: ATR Trailing in Active Trend

Situation: Long ETHUSDT 4H from 2800. ATR(14)=120. Trailing mult=2.0. Price has risen to 3200.

Answer: Trail_SL = 3200 − 2×120 = **2960 USD** (from entry 2800 = +5.7% guaranteed profit). If ETH rises to 3500: Trail_SL = 3500 − 240 = **3260 USD** (SL tightens, does not lower). Trailing_SL increases monotonically with price.

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

12. Cheat Sheet — Quick Summary

Concept	Value / Formula
TR	`max(H-L, |H-PC|, |L-PC|)`
ATR seed	`SMA(TR[1..period])` at bar period+1
ATR recursive	`TR[i]/N + ATR[i-1]×(N-1)/N`
K Wilder	`1/N` (NOT `2/(N+1)`)
Standard period	14
Warm-up	period+1 bars
Lag	~period bars (slow, Wilder)
SL Long	`Entry − mult × ATR`
SL Short	`Entry + mult × ATR`
Standard mult	1.5–2.0
Trailing Long	`max(prev_trail, Close − mult×ATR)`
Trailing Short	`min(prev_trail, Close + mult×ATR)`
Position sizing	`(Capital × Risk%) / (mult × ATR)`
ATR%	`ATR / Close × 100`
Increasing ATR	Expanding volatility
Decreasing ATR	Volatility compression (pre-squeeze)
Cause #1 divergence	K=2/(N+1) instead of K=1/N (Wilder)
Cause #2 divergence	Recursive EMA seed instead of SMA

PSEDYS ATR Technical Manual v1.0 — May 2026

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS

Downloaded from PSEDYS Ecosystem | t.me/PSEDYS