

Algorithme Ichimoku + ATR — Doc “phase-aware” halving & seeds HSBC v2025-08-21

1) Vue d'ensemble *pipeline<sub>web</sub>*

- Données: Binance via ccxt, timeframe 2h, cache local data/  
*CSV* *parsymbole*, *contrôlesqualité/gaps/volumes*.
- Stratégie cœur: Ichimoku *Tenkan/Kijun/SenkouB* + *shift* + trailing stop ATR.
- Entrées:
- Long: croisement Tenkan > Kijun ET Close > Nuage (après shift).
- Short: croisement inverse ET Close < Nuage (après shift).
- Sorties: croisement inverse OU trailing ATR
- Long:  $TS_t = \max(TS_{t-1}, Close_t - m \cdot ATR_t)$
- Short:  $TS_t = \min(TS_{t-1}, Close_t + m \cdot ATR_t)$
- Période ATR =  $\max(14, Kijun)$ ;  $m = atr_{mult}$ .
- Exécution réaliste: frais, funding/rollover, slippage dynamique *taille/volume*, latence simulée, haltes sur gaps extrêmes, limites & marges Binance.
- Optimisation: Optuna *TPE* + *ASHA* et/ou essaim génétique; walk-forward annuel; score multi-critères *Sharpe/CAGR/MaxDD/Stabilité*.
- Risque *réf.*: *position\_size*  $\approx$  1% de l'equity, levier  $\approx$  10×, max 3 positions par côté/symbole, stop global.
- Seed: --seed N fige random, numpy et le sampler TPE → reproductibilité des essais *pasdesdonnées*.
- Nouveau: module “phase” aligné halving BTC pour guider seeds et cadence des pools *ordonnanceurHSBC*.

2) Paramètres Ichimoku & logique

- Formules:  $Tenkan_N = (HH_N + LL_N/2)$ ;  $Kijun_M = (HH_M + LL_M/2)$ ;  $SenkouB_K = (HH_K + LL_K/2)$ , tracé en avance de shift périodes.
- Filtres optionnels: Chikou au-dessus/au-dessous du prix et du nuage; pente du nuage *SSA/SSB* positive/négative; MTF *H4/D1* permissifs.

Paramètres — effets clés:

Paramètre	Rôle	Effet clé	Ordres utiles
tenkan	signal rapide	sensibilité aux pullbacks/entrées	6–12
kijun	filtre tendance	inertie, retards vs faux signaux	26–100

Paramètre	Rôle	Effet clé	Ordres utiles
senkou_b	base nuage lente	épaisseur/structure du nuage	52–200
shift	avance nuage	synchronisation du filtre nuage	26–30
atr_mult	largeur du trail	tolérance au bruit vs whipsaw	1.5–5.0

Note: ATR période =  $\max(14, Kijun)$  → plus Kijun est long, plus l’ATR est lisse → trailing plus « calme ».

### 3) Module “phase” aligné halving Misplaced &

#### 3.1 Ancrage & features

- Halving:  $t=0$  = date du dernier halving.
- $H_{buy}$ : plus haut de clôture sur  $[-90j, +30j]$  autour de  $t=0$ .
- Ratio live:  $R_t = Close_t / H_{buy}$ .
- Features roulantes *sanslook-ahead*:
- Momentum  $M = EMA_{20} / EMA_{100} - 1$
- Vol annualisée  $V = \sigma(\logret_{30} \cdot \sqrt{365})$
- Drawdown  $DD = (P_t - \max(P_{t-365..t}) / \max(\cdot))$

#### 3.2 Règles heuristiques de phase à calibrer WF

- Accumulation:  $M \geq 0, V < 0.5, DD > -0.25$
- Expansion:  $M > 0.10, V \geq 0.5, DD > -0.20$
- Euphorie:  $M > 0.25, V \geq 0.8$
- Distribution:  $M < 0.10$  ou  $DD < -0.10$  avec  $V$  élevé
- Bear:  $M < 0$  et  $DD \leq -0.35$

#### 3.3 Cadence des pools HSBC par $R_t$

Bande $R_t$	Cadence pools	Idée
< 1.30	faible	prudence, peu d’explo, seeds « Accu »
1.30–1.60	moyenne	3–4 seeds/jour, « Expansion »
1.60–2.00	élevée	5–8 seeds/jour, entrées sur PB Tenkan/Kijun only
$\geq 2.00$	exploitation	réduire l’exploration, TP partiels, trails dynamiques

### 3.4 Seeds suggérées par phase *exemplesdetuples*

- Accumulation: 6, 26, 52, 26, 1.8, 7, 34, 60, 26, 1.5, 9, 43, 70, 26, 2.0
- Expansion: 6, 43, 100, 26, 3.0, 7, 55, 120, 30, 2.8, 9, 65, 120, 26, 3.5
- Euphorie: 6, 55, 120, 26, 4.0, 7, 65, 150, 26, 4.5, 9, 80, 200, 30, 5.0
- Distribution: 9, 65, 120, 26, 2.0, 10, 80, 150, 30, 2.5, 12, 100, 200, 30, 3.0
- Bear: 6, 26, 100, 26, 3.0, 7, 34, 150, 26, 3.5, 9, 55, 200, 30, 4.0

Plug-and-play: le scheduler HSBC lit phase +  $R_t$  et restreint l'espace des seeds; il peut muter  $\pm 10\text{--}20\%$  si l'exploit local est faible.

## 4) Gestion du risque *opérationnel*

- Sizing: risque/trade  $\approx 1\%$  de l'equity. Avec  $10\times$  de levier, viser un notionnel  $\approx 10\%$  de l'equity.
- Stops: toujours posés  $ATR$ . Stop global sur MaxDD intramensuel *ex.*  $-12$  coupe l'ordonnanceur.
- Concentration: max 3 positions par côté/symbole; si multi-paires, surveiller corrélations.
- Garde-fous de régime: si  $V_{30} \uparrow$  fort ou  $DD_{14} < \text{seuil}$   $\rightarrow$  ralentir cadence des pools même si  $R_t$  élevé.

## 5) Métriques & scoring

- Métriques: multiplicateur final  $equity_{mult}$ , CAGR, MaxDD, Calmar, Sharpe/Sortino, VaR95, proxy stabilité *Lyapunov*.
- Score *exemple*:  

$$Score = 0.35 \cdot Sharpe + 0.25 \cdot Calmar + 0.20 \cdot CAGR - 0.20 \cdot |MaxDD|$$
- Variantes: pénalités si instabilité inter-seeds ou si performance phase Bear  $<$  plancher.

## 6) Seed & reproductibilité

- Ce que contrôle `--seed`: `random.seed(seed)`, `np.random.seed(seed)`, sampler TPE Optuna *ordred'essais*, *pruningASHA*, *populationsgénétiques*.
- Ce que `--seed` ne contrôle pas: les données *marchéévolutif* et les dépendances si versions différentes.
- Quand changer de seed: tests de variance inter-seeds; plateau d'Optuna; réplication exacte d'un run (même seed + mêmes versions + même cache data/).
- Seeds conseillés: 42, 123, 777, 999.

## 7) Commandes CLI *prêtesàlancer*

```
# 1) Python – optimisation massive
python .\ichimoku_pipeline_web_v4_8_fixed.py pipeline_web6 --trials 5000 -

# 2) PowerShell – profil complet + baseline
pwsh -NoProfile -File .\run_full_analysis.ps1 -ProfileName pipeline_web6 -

# 3) PowerShell – série ciblée, label explicite
pwsh -NoProfile -File .\run_seed_python.ps1 -ProfileName pipeline_web6 -Tr
```

## 8) Validations robustesse 6checks

1) Walk-forward annuel IS/OOS *freezedesparamsparphase; aucune fuite d'info*. 2) Monte Carlo *blockbootsurordres* → IC sur CAGR/MaxDD/Sharpe. 3) Stress coûts: × 1.5–3 fees+slippage, latence ↑, haltes news → edge survivant ? 4) Stabilité inter-seeds: seed ∈ {42,123,777,999} → variance métriques < seuils. 5) Sensibilité hyperparams: heatmaps *kijun, atr<sub>mult</sub> & tenkan, shift* → plateau vs « aiguilles ». 6) Changement de régime: forcer Accu→Expan→Euph→Distrib→Bear et valider commutation *seeds, cadence, stops* sans drift.

## 9) Limites & risques

- Overfitting aux régimes passés; 2024+ peut diverger *multiplepost-halvingnongaranti*.
- Liquidité/slippage en vol extrême *news, week-end*; funding sur perpétuels.
- Dépendance au levier *risqueliquidationencascade*.
- Biais d'implémentation: look-ahead, mauvais shift du nuage, arrondis sur fees.
- Paramètres trop agressifs: *atr\_mult* trop bas en euphorie → whipsaws; *kijun* trop court → sur-réactivité.

## 10) Checklist avant live

- Cache data/ gelé & hashé; versions Python/libs/ccxt/Optuna figées.
- Backtests IS/OOS OK; WF validé; MC p50/p5 acceptables.
- Variance inter-seeds sous seuil; baseline archivée (BEST\_BASELINE.json).

- Phase engine:  $H_{buy}$ ,  $R_t$ ,  $M$ ,  $V$ ,  $DD$  calculés correctement *testunitaire*.
- HSBC: cadence pools conforme aux bandes  $R_t$ ; seuils d'arrêt  $DD/Vol$  actifs.
- Risque: sizing 1%, stop global, max 3 positions/côté/symbole; garde-fous news/gaps.
- Rapports: exports CSV/JSON/PNG + logs + labels incluant phase & seed.

## 11) Snippet YAML *plagesparphase*

```

phase_config:
  accumulation:
    seeds: [[6,26,52,26,1.8],[7,34,60,26,1.5],[9,43,70,26,2.0]]
    pool_intensity: low
    gates: {M_min: 0.0, V_max: 0.5, DD_min: -0.25}
  expansion:
    seeds: [[6,43,100,26,3.0],[7,55,120,30,2.8],[9,65,120,26,3.5]]
    pool_intensity: medium
    gates: {M_min: 0.10, V_min: 0.5, DD_min: -0.20}
  euphoria:
    seeds: [[6,55,120,26,4.0],[7,65,150,26,4.5],[9,80,200,30,5.0]]
    pool_intensity: high
    gates: {M_min: 0.25, V_min: 0.8}
  distribution:
    seeds: [[9,65,120,26,2.0],[10,80,150,30,2.5],[12,100,200,30,3.0]]
    pool_intensity: medium
    gates: {M_max: 0.10, DD_max: -0.10}
  bear:
    seeds: [[6,26,100,26,3.0],[7,34,150,26,3.5],[9,55,200,30,4.0]]
    pool_intensity: low
    gates: {M_max: 0.0, DD_max: -0.35}
  R_bands:
    low: [0.0, 1.30]
    mid: [1.30, 1.60]

```

```
high: [1.60, 2.00]
```

```
peak: [2.00, 99.0]
```

## 12) Pseudocode ordonnanceur HSBC *phase-aware*

```
phase = detect_phase(M, V, DD)
R = close / H_buy
seed_list = phase_config[phase]['seeds']
intensity = choose_intensity(R)
if risk_gates_violated(V30, DD14):
    intensity = downgrade(intensity)
candidates = mutate_if_needed(seed_list, pct=0.2) if underperforming else
schedule_pools(intensity, candidates)
```

## 13) Export du document en PDF *optionnel*

- Pré-requis: Microsoft Edge ou Google Chrome.

```
python .\scripts\export_docs_to_pdf.py --docs .\docs\HSBC_PHASE_HALVING_SE
```

Réfs internes: docs/USAGE.md, docs/HSBC\_REPORT\_FR.md,  
docs/FORMULES\_ET\_EXEMPLES.md, LOGIQUE\_PROGRAMME.md.