

网格策略的研究

1. 网格策略的核心思想（总纲）

网格策略本质上是：

在一个价格区间内，以分批买卖捕捉波动收益，同时控制风险，避免择时依赖。

它的有效性取决于三个关键点：

- 趋势环境是否适合网格（Regime Filter）
- 区间是否合理（支撑 / 阻力 / 波动率判断）
- 网格内部的仓位管理（Grid Position Management）

2. 趋势判断：是否适合网格？（Regime Detection）

网格策略不是万能策略，必须在正确的市场环境下运行。

1.1 适合网格的市场

- 震荡（Mean-reversion）
- 中性/弱趋势
- 波动率趋稳、有 clear 上下限

1.2 不适合网格的市场

- 大趋势（趋势突破 → 网格容易被单边吃穿）
- 极端波动、高杠杆清算潮
- 明显的“资金强烈单边流入/流出”

1.3 趋势过滤器

可考虑：

指标类别	用法
趋势强度（ADX、MACD slope、EMA 排列）	ADX > 25 → 禁止网格（趋势过强）
动能指标（ROC、Z-score）	动能正负明显 → 禁止网格
波动率状态（ATR、BB宽度）	过高波动率 → 降低网格密度或暂停

1.4 大盘的Regime 判断逻辑

1.4.1 GREEN 候选（主多环境）

1). 1 日线条件：

代码块

```
1 dev_D = math.abs(close_D - ema200_D) / ema200_D
2 slope_D = ema200_D - nz(ema200_D[20]) // 200EMA 20 日斜率
3 green_daily = (close_D > ema200_D)
4 and (ema50_D > ema200_D)
5 and (slope_D > 0)
6 and (dev_D < max_green_dev)
```

2). 4H 条件：连续若干根 4H 收盘 > 4H 200EMA：

代码块

```
1 var int green_4h_count = 0
2 green_4h_count := close_4H > ema200_4H ? green_4h_count + 1 : 0
3 bool green_4h_ok = green_4h_count >= green_4hBars_req
4 bool cond_green_candidate = green_daily and green_4h_ok
```

3). 4H 条件：按日线统计连续天数并做确认

代码块

```
1 isNewDay = ta.change(time("D")) != 0
2 var int green_days = 0
3
4 if isNewDay
5     green_days := cond_green_candidate ? green_days + 1 : 0
```

1.4.2 RED 候选（主空 / 高风险）

1) 熊市型 RED：

代码块

```
1 slope_D_down = ema200_D - nz(ema200_D[20])
2 red_bear = (close_D < ema200_D) and (ema50_D < ema200_D) and (slope_D_down < 0)
```

2). 疯牛末端型 RED：

```

1  red_bull = (close_D > ema200_D * (1 + red_bull_dev)) and (atr_D / close_D >
    atr_vol_thresh)
2  bool cond_red_candidate = red_bear or red_bull

```

3). 按日线统计：

代码块

```

1  var int red_days = 0
2  if isNewDay
3      red_days := cond_red_candidate ? red_days + 1 : 0

```

1.4.3 锁定机制与 Regime 状态变量

使用日线 time 计算锁定（防止频繁切换）：

代码块

```

1  var string regime = "YELLOW"
2  var int last_switch_day_time = time("D")
3  can_switch = time("D") - last_switch_day_time >= lock_days * 24 * 60 * 60 *
    1000
4
5  if isNewDay and can_switch
6      if green_days >= green_confirm_days
7          regime := "GREEN" last_switch_day_time := time("D")
8      else if red_days >= red_confirm_days
9          regime := "RED" last_switch_day_time := time("D")
10     else regime := "YELLOW"
11     last_switch_day_time := time("D")

```

3. 网格区间的确定（支撑/阻力 & 波动率）

网格最关键的是区间，区间选得好，策略自然收益的好。

2.1 区间来自哪？

你可以结合：

- 技术分析：支撑/阻力（4H、1H）
- 成交密集区（Volume Profile）
- 波动率带（布林带、Keltner Channel）
- 历史分布（Price Distribution / Percentile Bands）

2.2 建议的确定方式

结合 2 条线：

- **上界 (resistance zone)**
来自 4H 或 1H 关键供应区
- **下界 (support zone)**
来自需求区 / 成交密集区

如果以后希望更系统化： $\text{区间} = \text{中值} \pm k \times \text{波动率 (ATR / historical volatility)}$ 适合自动化 & 数字化。

4. 网格区间内的仓位管理（核心 Alpha 来源）

大多数业余网格策略“只靠波动”，但专业网格策略靠仓位管理来提升收益比和降低风险。

3.1 仓位不是平均分配，而是衰减式（Martingale 反向）

比如：

- 底部区域仓位更重
- 越往上越轻
- 避免在顶部堆满子弹

3.2 动态仓位（根据趋势过滤器）

例如：

- 若趋势偏多 → 网格偏向多单，减少空单
- 若趋势偏空 → 网格偏向空单
- 若趋势强 → 降低整体资金占用，甚至暂停

3.3 “爆仓价 / 清算风险” 控制

提前计算：

- 清算价
- 强平级别
- 满仓情况下的最大损失
- 多网格之间的资金占比

你的 `/grid_suggested` 模块就应该输出这类信息。

3.4 单格收益率 & 目标收益

在 neutral 区间网格，单格收益率决定策略利润上限：

单格收益 = (grid spacing / mid price) × position fraction

策略文档可定义：

- 默认 spacing = 1-2%
- 默认仓位 = 资金的 3-10%
- 最大资金暴露 = 30-50%

5. 风险控制（专业网格的差异点）

这是大多数网格新手最缺的。

4.1 多层风险管理

1. 区间失效止损（最重要）

下破支撑 = 停止加仓并整体止损

上破阻力 = 停止网格、等待回归

2. 时间止损

长期不波动 → 资金占用不合理

3. 资金占用限制

杠杆网格一定要严格控制 Max Exposure