

CryptoOracle 配置文件详解 (Configuration Guide)

本文档详细解析 `config.json` 的各项参数含义，帮助您正确配置现货与合约的混合交易策略。

1. 交易所配置 (exchanges)

配置连接交易所的 API 密钥。

🛡 安全升级 (v2.1)

为了保护您的资产安全，**不再推荐**将密钥直接明文写入 `config.json`。

请在项目 `src` 目录下创建 `.env` 文件来存储敏感信息（系统会自动读取）。

配置方法：

1. 进入 `src/` 目录。
2. 复制样本文件：`cp .env.example .env`
3. 编辑 `.env` 填入真实信息：

```
1 OKX_API_KEY=您的API_Key
2 OKX_SECRET=您的Secret_Key
3 OKX_PASSWORD=您的Passphrase
```

4. 在 `config.json` 中，您可以删除 `api_key` 等字段，或保留其为占位符（如 `YOUR_KEY_HERE`），系统会自动覆盖。

如何获取 API Key？

1. 登录 OKX 官网 -> 点击头像 -> API。
2. 点击 **创建 V5 API Key**。
3. 权限设置：必须勾选 "读取" 和 "交易"。
4. Passphrase：这是您创建 API 时设置的独立密码，**不是**您的登录密码。

```
1 "exchanges": {
2   "okx": {
3     // 密钥已移至 .env 文件管理，此处留空或填写占位符均可
4     "options": {} // 选填: ccxt 高级选项
5   }
6 }
```

2. 模型配置 (models)

配置用于市场分析的大语言模型。

同样推荐使用 .env 管理密钥：

在 `.env` 中添加：`DEEPSEEK_API_KEY=您的Key`

```
1 "models": {  
2     "deepseek": {  
3         // 密钥已移至 .env 文件管理  
4         "base_url": "https://api.deepseek.com", // API 地址  
5         "model": "deepseek-chat" // 模型名称 (建议保持默认)  
6     }  
7 }
```

3. 全局交易配置 (trading)

这里定义了所有机器人的默认行为。如果某个币种没有单独配置，就会使用这里的参数。

关于测试模式 (Test Mode)

- `"test_mode": true` (推荐初次使用):
 - 正常进行 AI 分析、计算指标。
 - 模拟下单，并在日志中显示“虚拟成交”。
 - 不会向交易所发送真实订单，不消耗任何资金。
- `"test_mode": false` (实盘):
 - 真实下单，盈亏自负。

```
1 "trading": {  
2     "market_type": "swap", // 默认市场类型。建议保持 "swap"，现货会自动兼容。  
3     "timeframe": "1m", // K线周期: "1m", "5m", "15m", "1h", "4h",  
4     "1d"  
5     "test_mode": true, // 测试模式开关  
6  
6     // --- 🌐 高级参数 (Advanced Settings) [新增] ---  
7     "proxy": "", // 代理地址 (如 "http://127.0.0.1:7890")，  
解决连接超时问题  
8     "max_slippage_percent": 1.0, // 最大滑点保护 (1.0 代表 1%): 防止分析后价格剧烈波动  
9     "min_confidence": "MEDIUM", // 最小信心门槛 (RESERVED): LOW / MEDIUM /  
HIGH  
10  
11     // --- 📈 策略参数 (strategy) ---  
12     "strategy": {  
13         "history_limit": 20, // [AI视角] 喂给 DeepSeek 分析的 K 线数量 (建议  
20-50)  
14         // 提示: 机器人后台会自动下载 100 根来计算技术指标  
(MACD等),  
15         // 但只把最近的 history_limit 根发给 AI, 既省钱又  
让 AI 专注近期趋势。  
16  
17         "signal_limit": 30, // [记忆容量] 机器人记住最近多少次 AI 的决策信号  
// 用于防止重复信号抖动和日志回溯。  
18  
19  
20         "dynamic_tp": true // [智能止盈] 是否启用动态止盈保护 (默认 true)  
// 开启后, 如果浮盈 > 1.5% 且动能衰竭, AI 会强制止盈。  
21     },  
22  
23     "margin_mode": "cross", // 默认保证金模式: "cross"(全仓) / "isolated"(逐  
仓)  
24     "trade_mode": "cross", // 默认交易模式: "cross"(合约) / "cash"(现货)  
25 }
```

```
26 // --- ⚡ 风险控制 (Risk Control) ---
27 "risk_control": {
28     "initial_balance_usdt": 100.0, // [必填] 初始本金: 用于计算盈亏比例
29
30     // --- 方案 A: 适合高手的绝对金额配置 (优先级高) ---
31     "max_profit_usdt": 0.0,           // 设为 0 代表禁用。
32     "max_loss_usdt": 15.0,          // [v2.1 推荐] 设为 15.0 代表亏 15U 止损 (抗
33     单模式)
34
35     // --- 方案 B: 适合小白的百分比配置 (推荐) ---
36     "max_profit_rate": 0.2,         // 0.2 代表盈利 20% 时止盈
37     "max_loss_rate": 0.15          // [v2.1 推荐] 放宽至 15% 以容忍剧烈波动
38 }
39 }
```

注意: 如果同时设置了 `usdt` 和 `rate`, 系统会**优先执行绝对金额 (usdt)**的限制。对于新手, 建议将 `usdt` 设为 0, 仅使用 `rate`。

4. 消息推送配置 (notification) [新增]

配置 Webhook 以接收实时交易通知。机器人会在以下情况发送消息:

- ⌚ 交易执行成功 (买入/卖出)
- ⚡ 止盈触发 / ⚡ 止损熔断
- ⚠ 下单失败或严重错误

支持平台:

- 钉钉 (DingTalk):** 自定义机器人 (安全设置选"自定义关键词", 填 `CryptoOracle` 或 `通知`)。
- 飞书 (Feishu):** 自定义机器人 Webhook。
- Slack / Discord:** Incoming Webhook URL。

推荐配置方式:

为了安全起见, 建议将 Webhook URL 配置在 `.env` 文件中, 不要直接写入 `config.json`。

1. 在 `.env` 中添加:

```
1 | NOTIFICATION_WEBHOOK=https://your_webhook_url_here
```

2. `config.json` 保持开启即可:

```
1 | "notification": {
2 |     "enabled": true
3 |     // "webhook_url": ""  <-- 留空或删除, 系统会自动读取环境变量
4 | }
```

5. 币种列表配置 (symbols) - 核心部分

这是实现**多币种、现货合约混跑**的关键。您可以在这里为每个币种定制策略。

关于资金分配 (allocation) [新增]

您可以通过此字段控制该币种可使用的资金额度。

- 百分比模式** (值 $<= 1.0$): 例如 `0.5` 代表使用总本金 (`initial_balance_usdt`) 的 50%。

- **固定金额模式** (值 > 1.0): 例如 `100` 代表固定分配 100 USDT 给该币种。
- **自动交易基准**: 当 `amount` 设为 `"auto"` 时, 机器人每次开仓默认使用此配额的 **10%**。
- **注意**: 这是一个**软限制**。如果您的账户余额不足, 实际下单数量会自动减少。

配置示例 A: 全仓永续合约 (BTC)

```
1 {  
2   "symbol": "BTC/USDT:USDT", // 格式注意: 合约必须带 :USDT 后缀  
3   "amount": "auto", // [推荐] 自动计算数量 (每次买配额的10%)  
4   "allocation": 0.5, // 资金分配: 使用 50% 的总本金  
5   "leverage": 10, // 杠杆倍数  
6   "trade_mode": "cross" // 模式: cross 代表全仓合约  
7 }
```

配置示例 B: 现货交易 (ETH)

```
1 {  
2   "symbol": "ETH/USDT", // 格式注意: 现货没有后缀  
3   "amount": 0.1, // 单次下单数量 (ETH)  
4   "allocation": 0.3, // 资金分配: 使用 30% 的总本金  
5   "leverage": 1, // 现货杠杆无效, 填 1 即可  
6   "trade_mode": "cash" // 模式: cash 代表现货交易  
7 }
```

⚠ [重要] 如何设置正确的 amount (数量)?

您有两种方式来设置每次交易的数量:

方案 A: 自动计算 (推荐) [新增]

直接将 `amount` 设置为 `"auto"`。

- **逻辑**: 机器人会自动计算。默认每次开仓使用该币种总分配额度 (`allocation`) 的 **10%**。
- **优势**: 省去手动计算的麻烦, 且会自动适配交易所的最小下单数量限制 (Minimum Order Size), 防止被拒单。

```
1 "amount": "auto"
```

方案 B: 手动指定

手动填写具体的“币的个数”(Quantity), 绝对不是 USDT 金额。

如果填错, 可能会导致下单失败 (小于最小限制) 或买入过多 (余额不足)。

正确计算公式:

`amount` = 您想投入的USDT金额 ÷ 币种当前价格

举例说明 (假设预算 30 USDT):

1. DOGE/USDT (当前价 \$0.13):

- 计算: $30 \div 0.13 \approx 230$ 个
- 推荐设置: `"amount": 230`

2. BTC/USDT (当前价 \$60000):

- 计算: $30 \div 60000 \approx 0.0005$ 个
- 推荐设置: `"amount": 0.0005`

提示: OKX 现货最小下单金额通常为 **5 USDT**。如果您的配置小于此值，机器人(在自动模式下)会尝试自动增加到最小限制。

配置示例 C：逐仓永续合约 (SOL)

```
1  {
2    "symbol": "SOL/USDT:USDT",
3    "amount": 1.0,
4    "leverage": 5,
5    "trade_mode": "isolated", // 模式: isolated 代表逐仓合约
6    "margin_mode": "isolated" // 保证金也需设为逐仓
7 }
```

常见问题 Q&A

Q1: market_type 是 "swap"，我的现货还能跑吗？

A: 可以。`ccxt` 库非常智能，当你请求 `ETH/USDT` 这种标准现货符号时，它会自动切换到底层现货接口，忽略默认的 swap 设置。

Q2: 如何从测试切换到实盘？

A: 将 `trading` 下的 `"test_mode": true` 改为 `false`。建议先用最小金额（如 `0.01 BTC`）跑 24 小时观察稳定性。

Q3: 为什么余额显示为 0？

A: 请检查您的 USDT 是否在“**交易账户**”(Trading Account)。

- OKX 分为“资金账户”和“交易账户”。
- 机器人只能读取和操作“交易账户”里的钱。
- 请在 OKX App 内进行“资金划转”：资金账户 -> 交易账户。

Q4: 为什么机器人一直 HOLD 不交易？

A: 这通常是好现象！

- 机器人配置了严格的 AI 过滤（盈亏比 > 1.1 ）和 ADX 趋势确认。
- 在震荡或无方向的市场中，**空仓也是一种策略**。
- 您可以尝试将 `config.json` 中的 K 线周期 `timeframe` 改为更短（如 `1m` 或 `5m`）来增加交易频率。

Q5: 报错 Invalid API Key 怎么办？

A:

1. 检查 `api_key`, `secret`, `password` 是否复制完整（不要有多余空格）。
2. 检查 IP 白名单：如果您在云服务器跑，需要把服务器 IP 加到 OKX 后台。如果您是本地跑且 IP 变动，可以暂时不设白名单（注意安全）。
3. 检查是否选择了“模拟盘”API：本机器人默认连接 OKX 实盘 API。如果您申请的是模拟盘 Key，请修改代码中的 `exchange.set_sandbox_mode(True)`（需手动添加）。

Q6: history_limit 和 100 根 K 线的关系？

A:

- 机器人会在后台自动下载 **100 根** K 线，用于计算 RSI、MACD、ADX 等技术指标（指标需要足够的历史数据预热才准确）。
- `history_limit (默认20)` 控制的是**发给 AI 的 K 线数量**。我们只把最近 20 根发给 DeepSeek，让 AI 专注分析近期的价格行为，既节省 Token 费用，又避免 AI 被过远的历史走势干扰。

