

参数配置

- 参数配置

API 手册

- 入口函数
- 约定函数
- 交易接口
- 仓位查询接口
- 数据查询接口
- 其他接口
- scheduler 定时器
- 类
- 枚举常量

- »
- 参数配置
- View page source

参数配置1

本节所列参数配置用于传入 deepquant 的函数入口，或赋值给策略的 __config__ 全局变量。

这些配置项被分成两部分——Base（基础）部分、Mod（扩展模块）部分和 Extra（其他）部分，其中 Mod 部分又按功能划分为了多个模块。使用时，需按照如下结构构建字典。

如无特殊说明，下述字典中各项的值即为该配置项的默认值。

```
{
  "base": {
    # 数据包存放的路径
    "data_bundle_path": "~/yhalphaplus/bundle",
    # 策略运行开始日期
    "start_date": "2015-06-01",
    # 策略运行结束日期
    "end_date": "2050-01-01",
    # 保证金倍率，即在默认保证金率的基础上以该倍数进行调整
    "margin_multiplier": 1,
    # 回测频率，可选值："1d"|"1m"|"tick"
    "frequency": "1d",
    # 账户设置，默认为空；key 为账户类型，值为账户初始资金
    "accounts": {
      # 股票账户，下属资产包括股票、基金、可转债和沪深两市的期权等
      "STOCK": 1000000,
      # 期货账户，下属资产包括期货和各期货交易所的期权
      "FUTURE": 1000000,
    },
    # 初始化仓位, 格式参考 000001.SH:1000, IF1701:-1
    "init_positions": "",
    # yhdata 配置项，当不打算使用 yhdata 时可设置为 'disabled' 或 'DISABLED'
    "yhdatac_uri": "",
    # 是否开启期货历史交易参数进行回测
    "futures_time_series_trading_parameters": True,
    # 是否开启在回测过程中自动下载所需的 bundle 数据
    # 当前支持数据：1. 盘前集合竞价成交量
    "auto_update_bundle": True,
    # 自动下载的 bundle 文件支持单独设置存储路径，若不设置则使用 data_bundle_path 路径
    "auto_update_bundle_path": None
  },

  "extra": {
    # 系统日志级别，用于控制策略框架输出日志的详细程度（策略打印的日志不受该选项控制），设置为某一级别则框架会输出该级别及更“严重”的日志
    # 可选值："debug"|"info"|"warning"|"error"，通常推荐设置为 info 或 warning
    # error 日志一般为不可逆的错误，如策略抛出异常、加载 Mod 失败等
    # warning 日志一般为告警信息，如 API 废弃、订单创建失败等
    # info 日志一般为说明性的信息，如 Mod 在某种设置下被动关闭等
    # debug 日志一般为开发者关注的调试信息，如策略状态变更、事件触发等，用户通常不需要关注
    "log_level": "info",
    # 是否开启性能分析
    "enable_profiler": False,
    # 输出的日志文件路径
    "log_file": None,
  },

  "mod": {
    # 股票/期货模块，该模块的配置项用以控制股票期货品种的相关逻辑
    "sys_accounts": {
      # 是否开启股票 T+1 限制
      "stock_t1": True,
      # 是否开启自动分红再投资
      "dividend_reinvestment": False,
      # 当持仓股票退市时，是否按照退市价格返还现金
      "cash_return_by_stock_delisted": True,
```

```
# 股票下单因资金不足被拒时改为使用全部剩余资金下单
"auto_switch_order_value": False,
# 开启对股票仓位是否能满足平仓需求的检查
"validate_stock_position": True,
# 开启对期货仓位是否能满足平仓需求的检查
"validate_future_position": True,
# 融资利率/年
"financing_rate": 0.00,
# 是否开启融资可买入股票的限制
"financing_stocks_restriction_enabled": False,
# 逐日盯市结算价: settlement/close
"futures_settlement_price_type": "close",
},

# 风控模块, 该模块的配置项用以控制事前风控的相关逻辑
"sys_risk": {
    # 开启对限价单价格合法性的检查
    "validate_price": True,
    # 开启对标的可交易情况对检查
    "validate_is_trading": True,
    # 开启对可用资金是否足够满足下单要求的检查
    "validate_cash": True,
    # 开启对存在自成交风险的检查
    "validate_self_trade": False,
},

# 模拟撮合模块, 该模块的配置项用以设置模拟撮合等逻辑
"sys_simulation": {
    # 开启信号模式: 该模式下, 所有通过风控的订单将不进行撮合, 直接产生交易
    "signal": False,
    # 撮合方式, 其中:
    #   日回测的可选值为 "current_bar"|"vwap" (以当前 bar 收盘价|成交量加权平均价撮合)
    #   分钟回测的可选值有 "current_bar"|"next_bar"|"vwap" (以当前 bar 收盘价|下一个 bar 的开盘价|成交量加权平均价撮合)
    #   tick 回测的可选值有 "last"|"best_own"|"best_counterparty" (以最新价|己方最优价|对手方最优价撮合) 和 "counterparty_offer" (逐档撮合)
    #   matching_type 为 None 则表示根据回测频率自动选择. 日/分钟回测下为 current_bar, tick 回测下为 last
    "matching_type": None,
    # 开启对于处于涨跌停状态的证券的撮合限制
    "price_limit": True,
    # 开启对于对手盘无流动性的证券的撮合限制 (仅在 tick 回测下生效)
    "liquidity_limit": False,
    # 开启成交量限制
    #   开启该限制意味着每个 bar 的累计成交量将不会超过该时间段内市场上总成交量的一定比值 (volume_percent)
    #   开启该限制意味着每个 tick 的累计成交量将不会超过当前tick与上一个tick的市场总成交量之差的一定比值
    "volume_limit": True,
    # 每个 bar/tick 可成交数量占市场总成交量的比值, 在 volume_limit 开启时生效
    "volume_percent": 0.25,
    # 滑点模型, 可选值有 "PriceRatioSlippage" (按价格比例设置滑点) 和 "TickSizeSlippage" (按跳设置滑点)
    #   亦可自己实现滑点模型, 选择自己实现的滑点模型时, 此处需传入包含包和模块的完整类路径
    #   滑点模型类需继承自 yhalph.mod.yhalph_mod_sys_simulation.slippage.BaseSlippage
    "slippage_model": "PriceRatioSlippage",
    # 设置滑点值, 对于 PriceRatioSlippage 表示价格的比例, 对于 TickSizeSlippage 表示跳的数量
    "slippage": 0,
    # 开启对于当前 bar 无成交量的标的的撮合限制 (仅在日和分钟回测下生效)
    "inactive_limit": True,
    # 账户每日计提的费用, 需按照(账户类型, 费率)的格式传入, 例如[("STOCK", 0.0001), ("FUTURE", 0.0001)]
    "management_fee": [],
},

# 费用模块, 该模块的配置项用于调整交易的税费
"sys_transaction_cost": {
    # 股票最小手续费, 单位元
    "cn_stock_min_commission": 5,
    # 佣金倍率 (即将废弃)
    "commission_multiplier": 1,
    # 股票佣金倍率, 即在默认的手续费率基础上按该倍数进行调整, 股票的默认佣金为万八
    "stock_commission_multiplier": 1,
    # 期货佣金倍率, 即在默认的手续费率基础上按该倍数进行调整, 期货默认佣金因合约而异
    "futures_commission_multiplier": 1,
    # 印花倍率, 即在默认的印花税基础上按该倍数进行调整, 股票默认印花税为万分之五, 单边收取
    "tax_multiplier": 1,
    # 是否使用回测当时时间点对应的真实印花税率
    "pit_tax": False,
},

# 分析模块, 该模块的配置项用以控制策略运行结束后的指标计算和结果输出
"sys_analyser": {
    # 策略基准, 该基准将用于风险指标计算和收益曲线图绘制
    #   若基准为单指数/股票, 此处直接设置 market_code, 如: "000300.XSHG"
    #   若基准为复合指数, 则需传入 market_code 和权重构成的字符串, 如: "000300.XSHG:0.2,000905.XSHG:0.8"
    #   若希望使用某个指数一定比例的收益作为基准, 则可传入 market_code 和 null/NULL 权重构成的字符串 (null/NULL 表示 0 收益序列), 如:
    #       使用 000300.XSHG 的一半收益作为基准, 则可设置为: "000300.XSHG:0.5,null:0.5" 或 "000300.XSHG:0.5,NULL:0.5"
    "benchmark": None,
    # 当不输出 csv/pickle/plot 等内容时, 关闭该项可关闭策略运行过程中部分收集数据的逻辑, 用以提升性能
    "record": True,
    # 策略名称, 可设置 summary 报告中的 strategy_name 字段, 并展示在 plot 回测结果图中
    "strategy_name": None,
    # 回测结果输出的文件路径, 该文件为 pickle 格式, 内容为每日净值、头寸、流水及风险指标等; 若不设置则不输出该文件
    "output_file": None,
    # 回测报告的数据目录, 报告为 csv 格式; 若不设置则不输出报告
    "report_save_path": None,
    # 是否在回测结束后绘制收益曲线图
    'plot': False,
    # 收益曲线图路径, 若设置则将收益曲线图保存为 png 文件
    'plot_save_file': None,
    # 收益曲线图设置
    'plot_config': {
        # 是否在收益图中展示买卖点
        'open_close_points': False,
        # 是否在收益图中展示周度指标和收益曲线
        'weekly_indicators': False
    },
},
```

```
    },

    # 在回测运行过程中绘制进度条的模块
    "sys_progress": {
        # 是否在命令行/终端绘制进度条
        "show": False,
    },

    # 期权模块
    "option": {
        # 行权滑点，该参数用于调整自动行权的判定逻辑，不影响行权实际产生的损益
        # 即行权价与底层标的价格的差异达到标的价格的一定比例才会触发行权
        "exercise_slippage": 0,
        # 自定义品种手续费，默认为空的 dict，当设置了某品种又没指定相关的值时，默认取0
        # 案例：
        # "commission": {
        #     "ZN": {
        #         "open": 10, # 开仓手续费
        #         "close": 1, # 平仓手续费
        #         "exercise": 2, # 行权手续费
        #         "close_today": 3 # 平今手续费
        #     }
        # }
        "commission": {}
    },

    # 贵金属现货模块
    "spot": {
        # 贵金属现货佣金倍率，即在默认费率基础上按该倍数进行调整
        "commission_multiplier": 1
    },

    # 可转债模块
    "convertible": {
        # 可转债佣金费率
        "commission_rate": 0,
        # 可转债最小佣金
        "min_commission": 0,
        # 载入mod优先级
        "priority": 220,
    },

    # 公募基金模块
    "fund": {
        # 基金前端费率
        "fee_ratio": 0.015,
        # 基金申购份额到账天数
        "subscription_receiving_days": 1,
        # 基金赎回款到账天数
        "redemption_receiving_days": 3,
        # 是否开启申赎金额上下限限制
        "subscription_limit": True,
        # 是否开启申赎状态检查
        "status_limit": True,
        # 载入mod优先级
        "priority": 220,
        # 非货币基金净值类型 unit(单位净值) / adjusted(复权净值)
        "fund_nav_type": "unit",
    },

    # 增量回测模块
    "incremental": {
        # 是否启用 csv 保存 feeds 功能，可以设置为 MongoddbRecorder
        "recorder": "CsvRecorder",
        # 当设置为 CsvRecorder 的时候使用，持久化数据输出文件夹
        "persist_folder": None,
        # 当设置为 MongoddbRecorder 的时候使用
        "strategy_id": 1,
        "mongo_url": None,
        "mongo_dbname": "yhalpha_records",
        # 载入mod优先级
        "priority": 111,
    },

    # ams 模块：用于帮助上传回测流水
    "ams": {
        "enabled": True,
        # 上传的产品地址
        "ams_product": "https://username:password@www.ricequant.com/workspace/product",
        # 是否重置流水，重置表示删除start_date之后的流水再重新上传
        "reset_trades": True,
    }
}
```

[Next Previous](#)