Where Market Making Meets Market Microstructure

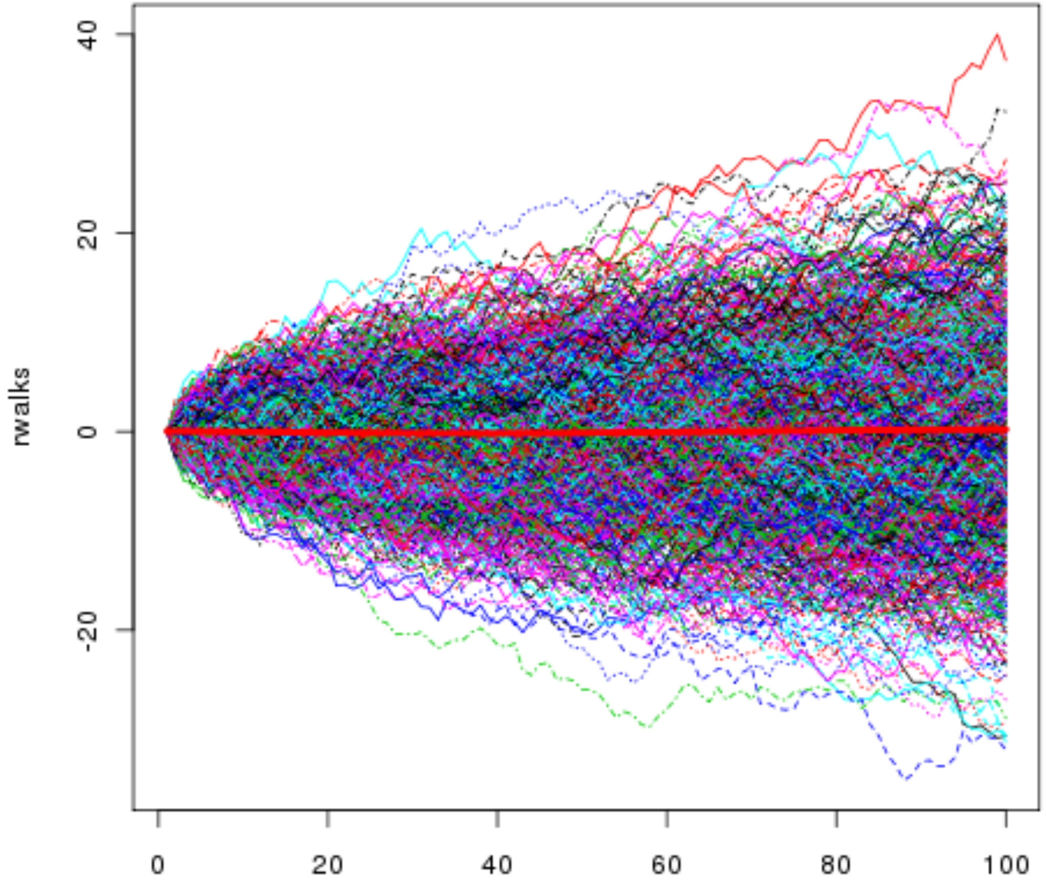
Quantitative Finance, Vol. 8, No. 3, April 2008, 217–224

Routledge
Taylor & Francis Group

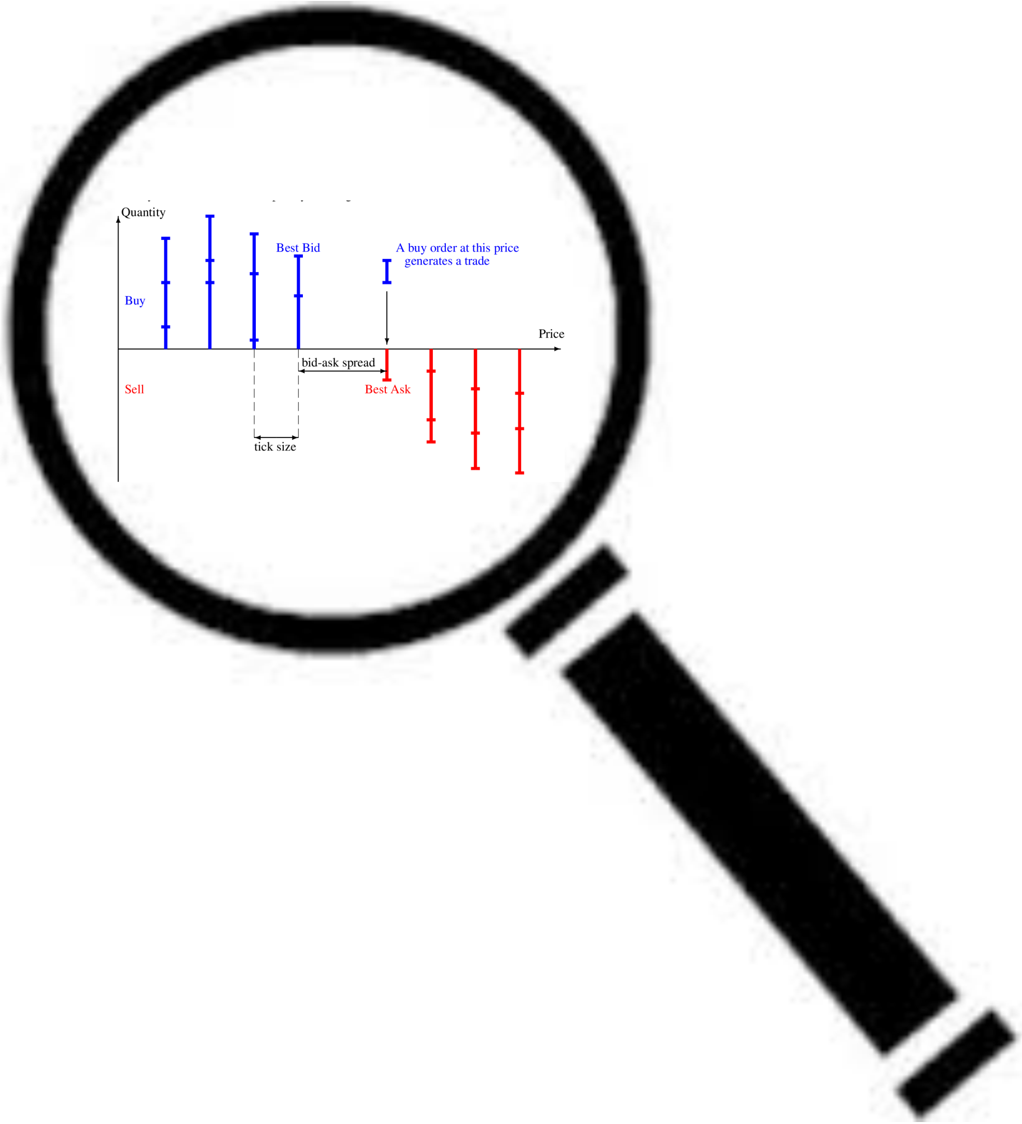
High-frequency trading in a limit order book

MARCO AVELLANEDA and SASHA STOIKOV*

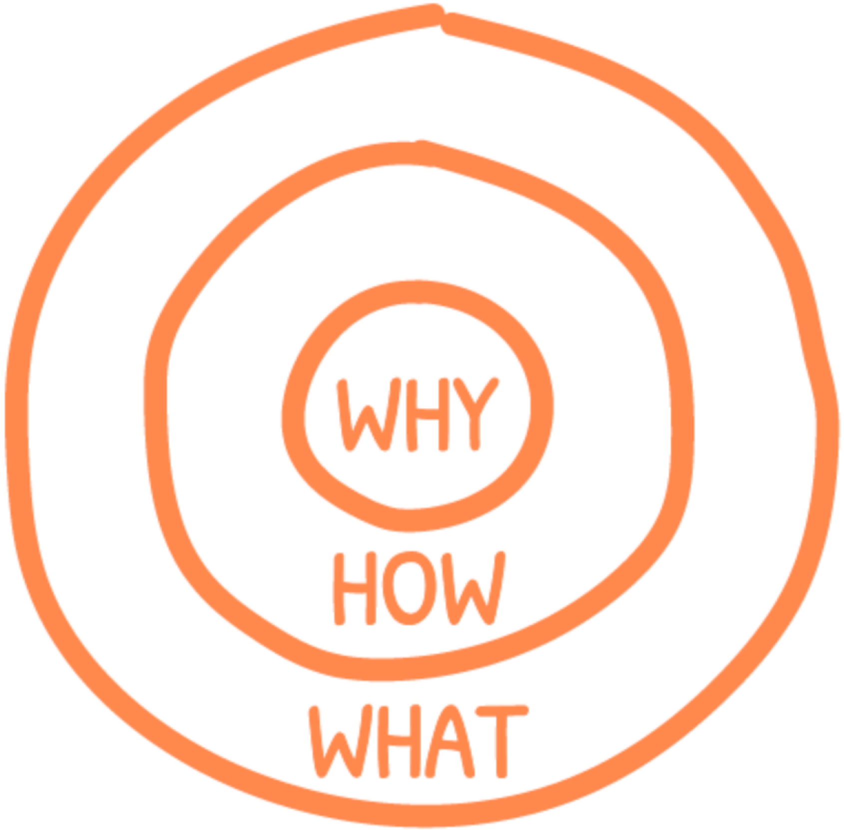
Market Making



Market Microstructure



Market Making



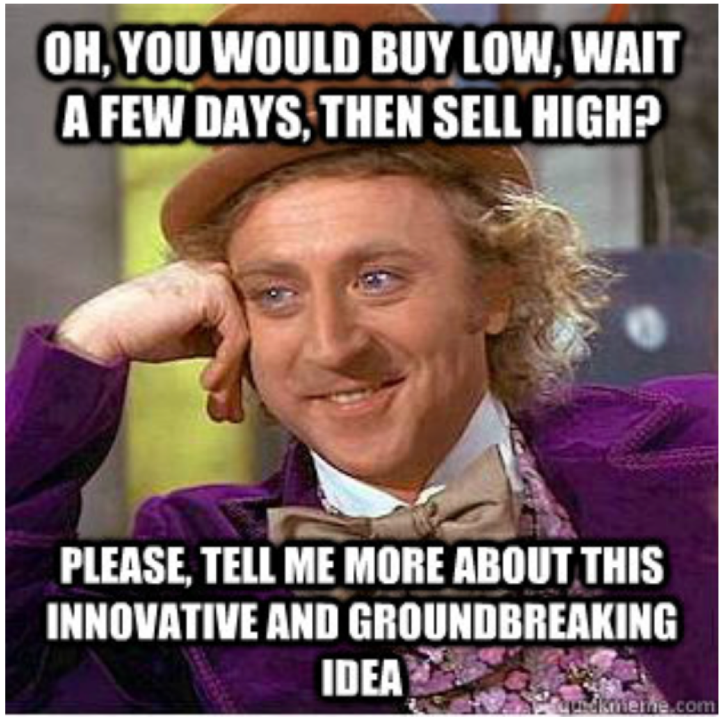

11

Academic literature on Market Making

- 1980: Ho and Stoll: general utility, monopolistic dealer
- 2008: Avellaneda and Stoikov: competitive order books, closed form
- 2012: Geant, Lehalle and Fernandez: bounded inventory
- 2014: Cartea, Jaimungal and Ricci: price impact and adverse selection
- 2017: Gueant: multi asset case
- 2022: Aydoğan, Uğur and Aksoy: stochastic volatility

Market Making

Idea



How do market makers make money?

- Making the bid/ask spread
- Quoting around the right price level
- Managing the inventory to reduce risk

What is the right price?

- The market midprice:

$$s$$

- The inventory adjusted level:

$$r = s - \beta q$$

- The optimal bid and ask quotes:

$$p^b = r - \delta$$
$$p^a = r + \delta$$

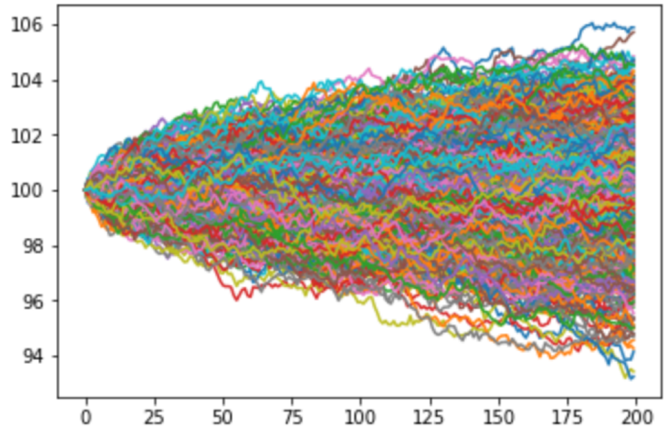
Market Making

Optimization

The mid price of the stock is Brownian motion

$$dS_t = \sigma dW_t$$

In [158]:
```
S = BM(T,dt,nsims)
plt.plot(S)
plt.show()
```



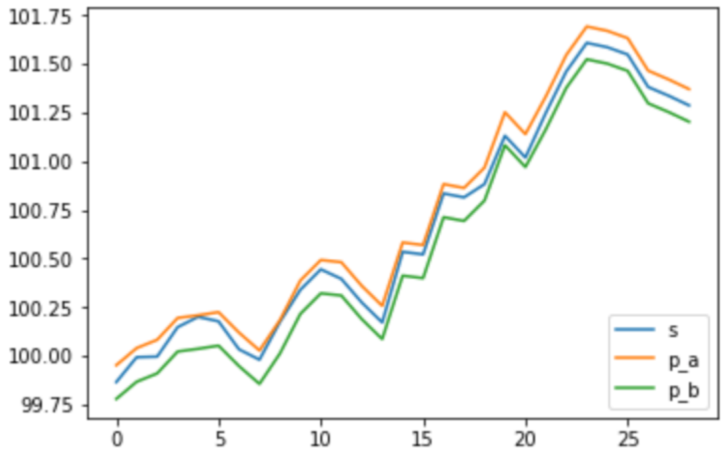
- The arrival of buy orders is Poisson with intensity λ^b

$$\lambda^b = A \exp(-k(s - p^b))$$

- The arrival of sell orders is Poisson with intensity λ^a

$$\lambda^a = A \exp(-k(p^a - s))$$

In [163]:
```
plt.plot(S[1:30,0],label='s')
plt.plot(p_a[1:30,0],label='p_a')
plt.plot(p_b[1:30,0],label='p_b')
plt.legend(loc='lower right')
plt.show()
```



The market maker's objective

Maximize exponential utility

$$u(s, x, q, t) = \max_{p^a, p^b} E_t \left[-e^{-\gamma(X_T + q_T S_T)} | S_t = s, X_t = x, q_t = q \right]$$

The wealth in cash

$$dX_t = p^a dN_t^a - p^b dN_t^b$$

The inventory

$$q_t = N_t^b - N_t^a$$

The Hamilton-Jacobi-Bellman equation

$u(x, s, q, t)$ solves

$$\begin{cases} u_t + \frac{1}{2}\sigma^2 u_{ss} \\ + \max_{p^b} \lambda^b(p^b) \left[u(s, x - p^b, q + 1, t) - u(s, x, q, t)\right] \\ + \max_{p^a} \lambda^a(p^a) \left[u(s, x + p^a, q - 1, t) - u(s, x, q, t)\right] = 0 \\ u(S, x, q, t) = -\exp(-\gamma(x + qS)). \end{cases}$$

The optimal quotes (approx)

- Step one: the indifference price
$$r(s, q, t) = s - q\gamma\sigma^2(T - t)$$
- Step two: the bid/ask quotes
$$p^b = r - \frac{1}{2}\gamma\sigma^2(T - t) - \frac{1}{\gamma}\ln\left(1 + \frac{\gamma}{k}\right)$$
and
$$p^a = r + \frac{1}{2}\gamma\sigma^2(T - t) + \frac{1}{\gamma}\ln\left(1 + \frac{\gamma}{k}\right).$$
k is a measure of the liquidity of the market.

Market Making

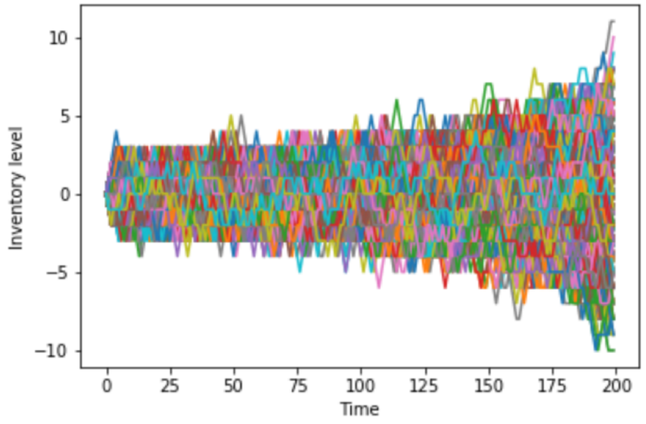
Simulation

1. The market maker's Inventory
2. The market maker's PNL

Inventory trajectories for the inventory strategy

Notice the inventory control is stricter at the beginning of a day

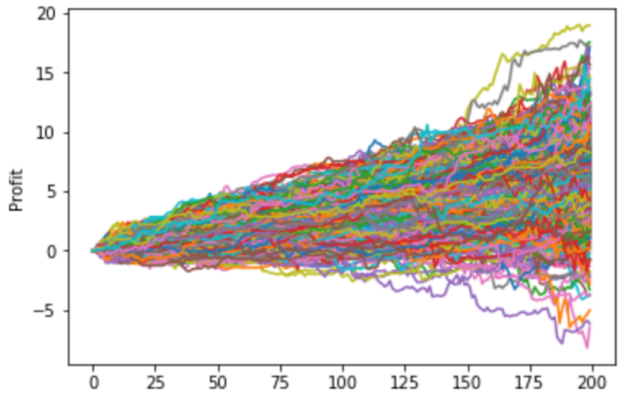
In [166]:
```
plt.plot(Q_inventory)
plt.xlabel('Time')
plt.ylabel('Inventory level')
plt.show()
```



Profit of the inventory strategy

In [28]:
```
plt.plot(profit_inventory)
plt.ylabel('Profit')
```

Out[28]: <matplotlib.text.Text at 0x7f9570fca198>

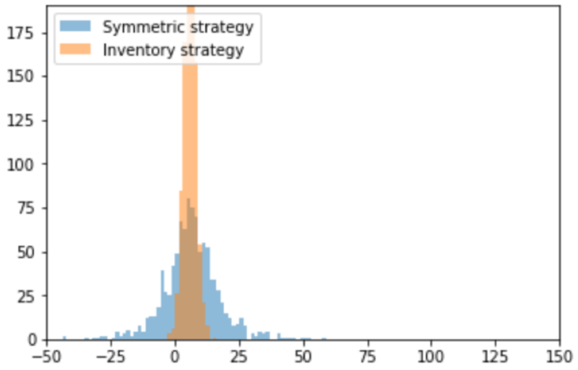


Histogram of profits

with and without inventory control

In [18]:

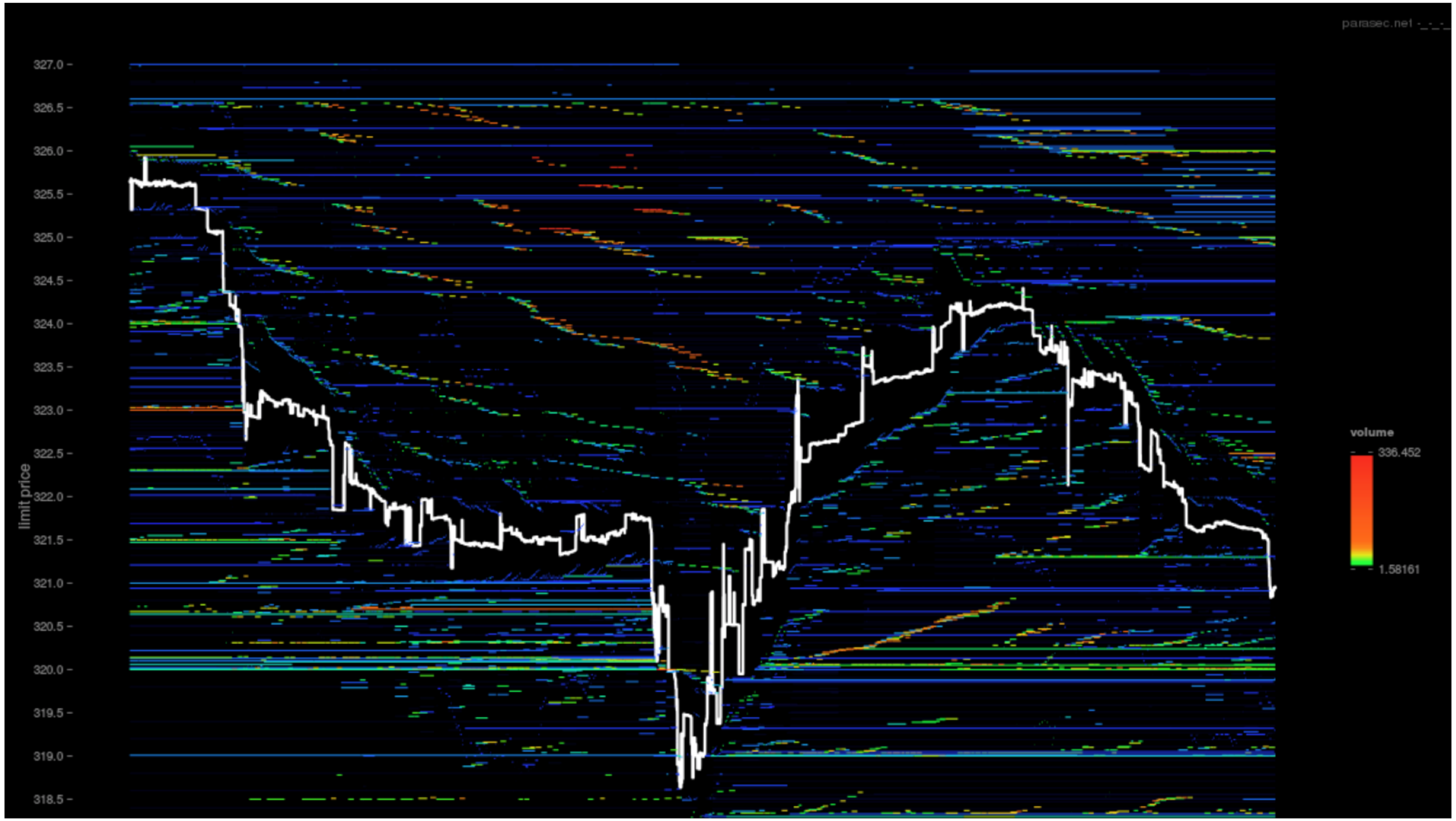
```
plt.xlim((-50,150))
plt.ylim((0,190))
n, bins, patches = plt.hist(profit_symmetric[-1,:],alpha=0.5,bins=70,label='Symmetric strategy
plt.hist(profit_inventory[-1,:],alpha=0.5,bins=bins,label='Inventory strategy')
plt.legend(loc='upper left')
plt.show()
```



Market Making

Live Trading

Running Live on Crypto



Hummingbot

HBOT Botcamp Developers Miner About Blog Test Drive

Open Source Market Making

Hummingbot is open source software that helps you build **high-frequency crypto trading bots** that specialize in market making and arbitrage strategies

Test Drive Learn with Botcamp

```
>>>  create

What is your market making strategy? >>> avellaneda_market_making
Please see https://docs.hummingbot.io/strategies/avellaneda-market-making/ while setting up the
se below configuration.

Input your maker spot connector >>> kucoin_paper_trade

Enter the token trading pair you would like to trade on kucoin_paper_trade (e.g. ETH-USDT) >>>
BTC-USDT

Select the execution timeframe (infinite/from_date_to_date/daily_between_times) >>> infinite

What is the amount of BTC per order? >>> 0.1

Enter risk factor (γ) >>> 2

How often do you want to cancel and replace bids and asks (in seconds)? >>> 10

What is the inventory target for the base asset? Enter 50 for 50% >>> 50

Enter a new file name for your configuration >>> conf_avellaneda_market_making_3.yml
A new config file has been created: conf_avellaneda_market_making_3.yml

Preliminary checks:
  - Strategy check: All required parameters confirmed.
  - Exchange check: All connections confirmed.
  - All checks: Confirmed.

Enter "start" to start market making.
```

Strategy Configurations:

Key	Value
strategy	avellaneda_market_making
exchange	kucoin_paper_trade
market	BTC-USDT
execution_timeframe_mode	infinite
order_amount	0.1
order_optimization_enabled	True
risk_factor	2
order_amount_shape_factor	0
min_spread	0
order_refresh_time	10.0
max_order_age	1800.0
order_refresh_tolerance_pct	0
filled_order_delay	60.0
inventory_target_base_pct	50
add_transaction_costs	False
volatility_buffer_size	200
trading_intensity_buffer_size	200
order_levels_mode	single_order_level
order_override	None
hanging_orders_mode	ignore_hanging_orders
should_wait_order_cancel_confirmation	True

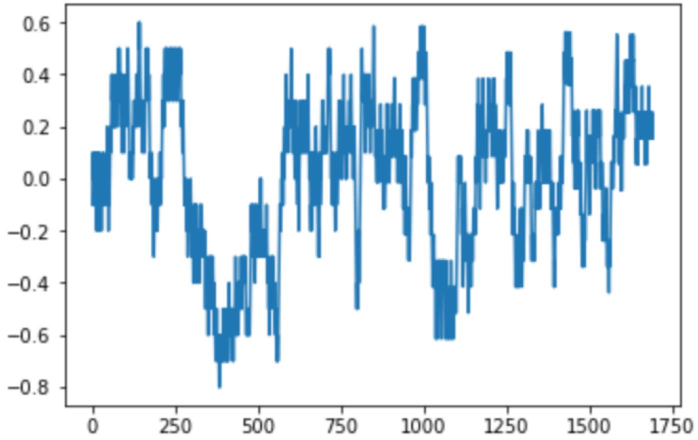
```
> status

Paper Trading Active: All orders are simulated, and no real orders are placed.

Markets:
            Exchange   Market  Best Bid  Best Ask  MidPrice Reservation Price Optimal Spread
  kucoin_PaperTrade BTC-USDT   22978.7   22978.8  22978.75       22976.22505        4.28982

Assets:
                                BTC       USDT
  Total Balance                 0.9 17295.5392
  Available Balance             0.9 14998.1312
  Current Value (USDT) 20680.875 17295.5392
  Current %                   54.5%      45.5%

Orders:
   Level Type    Price Spread  Amount (Orig)  Amount (Adj)      Age
       1  buy 22974.08  0.02%            0.1            0.1 00:00:07

Strategy parameters:
  risk_factor(γ)= 2.00000E+0
  order_book_intensity_factor(A)= 3.59787E+0
  order_book_depth_factor(κ)= 7.25608E+0
  volatility= 0.009%
  time until end of trading cycle = N/A
023-02-07 09:33:57) Maker BUY order 0.1000000 BTC @ 22974.0800000000 USDT is filled.
```

Good news

My inventory was under control

In [118]:
```
df_AS['inventory'].plot()
```

Out[118]: <matplotlib.axes._subplots.AxesSubplot at 0x7fce7b5d5160>

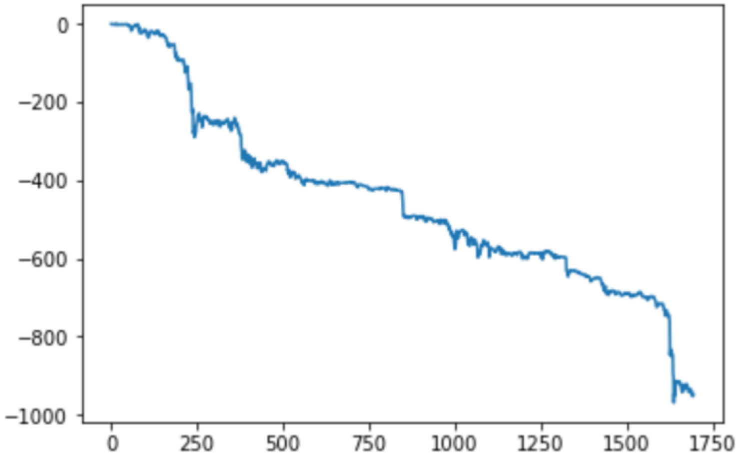


Bad news

The PNL was awful

In [117]:
```
df_AS['pnl'].plot()
```

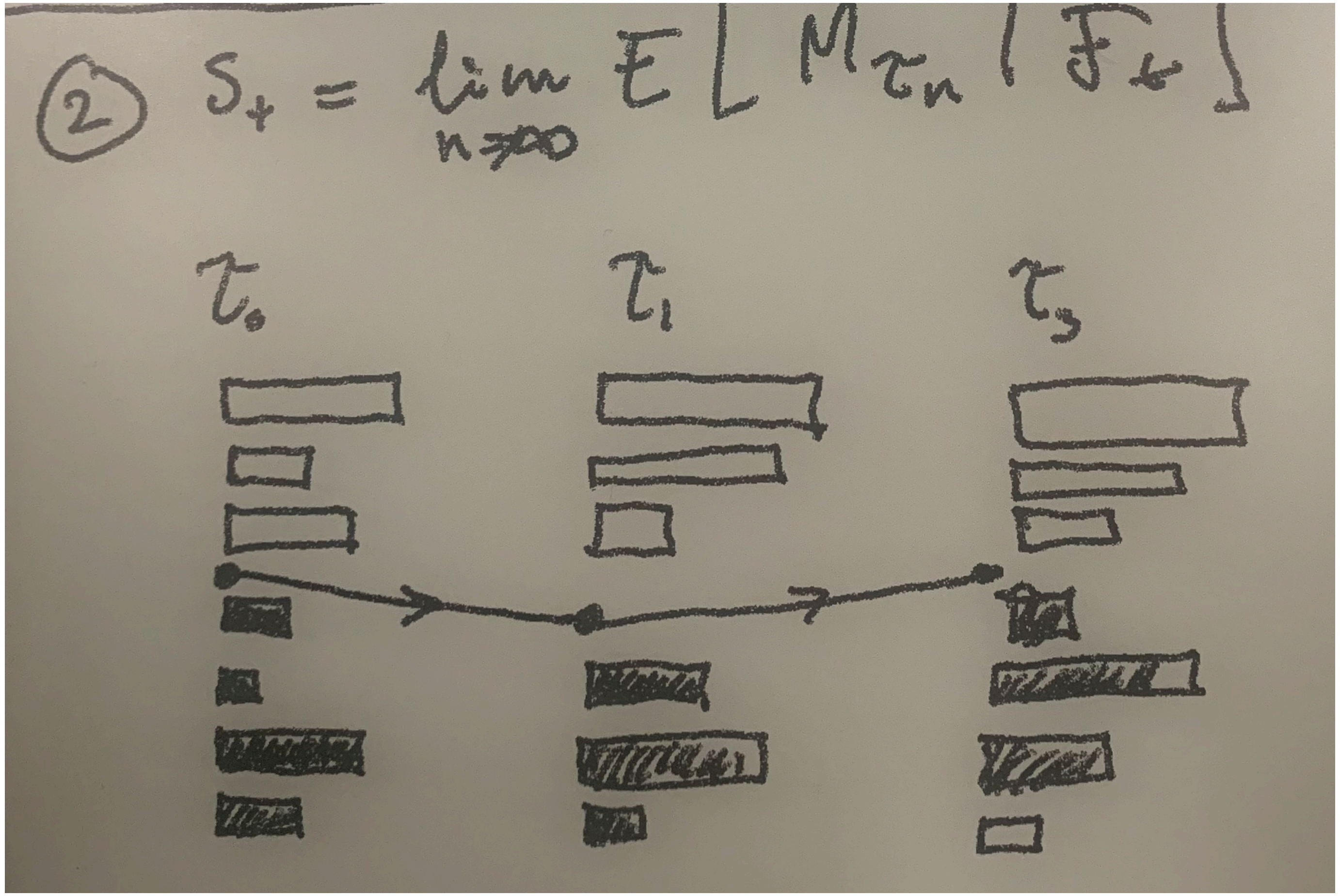
Out[117]: <matplotlib.axes._subplots.AxesSubplot at 0x7fce7b58b240>



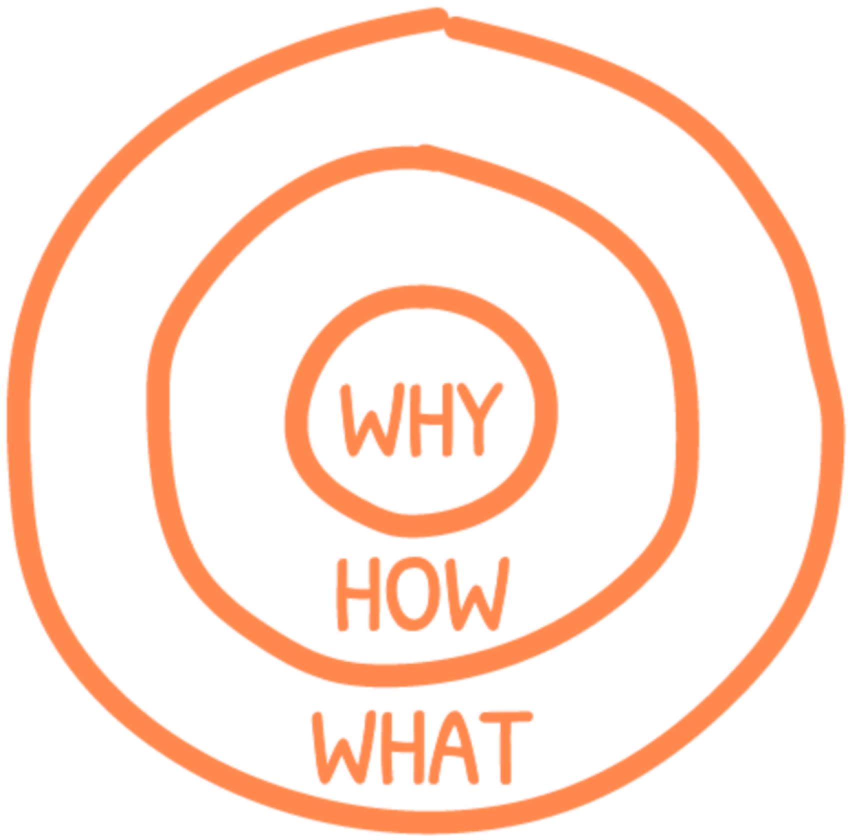
What happened?

- Price of bitcoin went down
- Parameters not fine tuned
- Inventory was kept under control
- The midprice has no alpha

Is the midprice the right price?



Market Microstructure



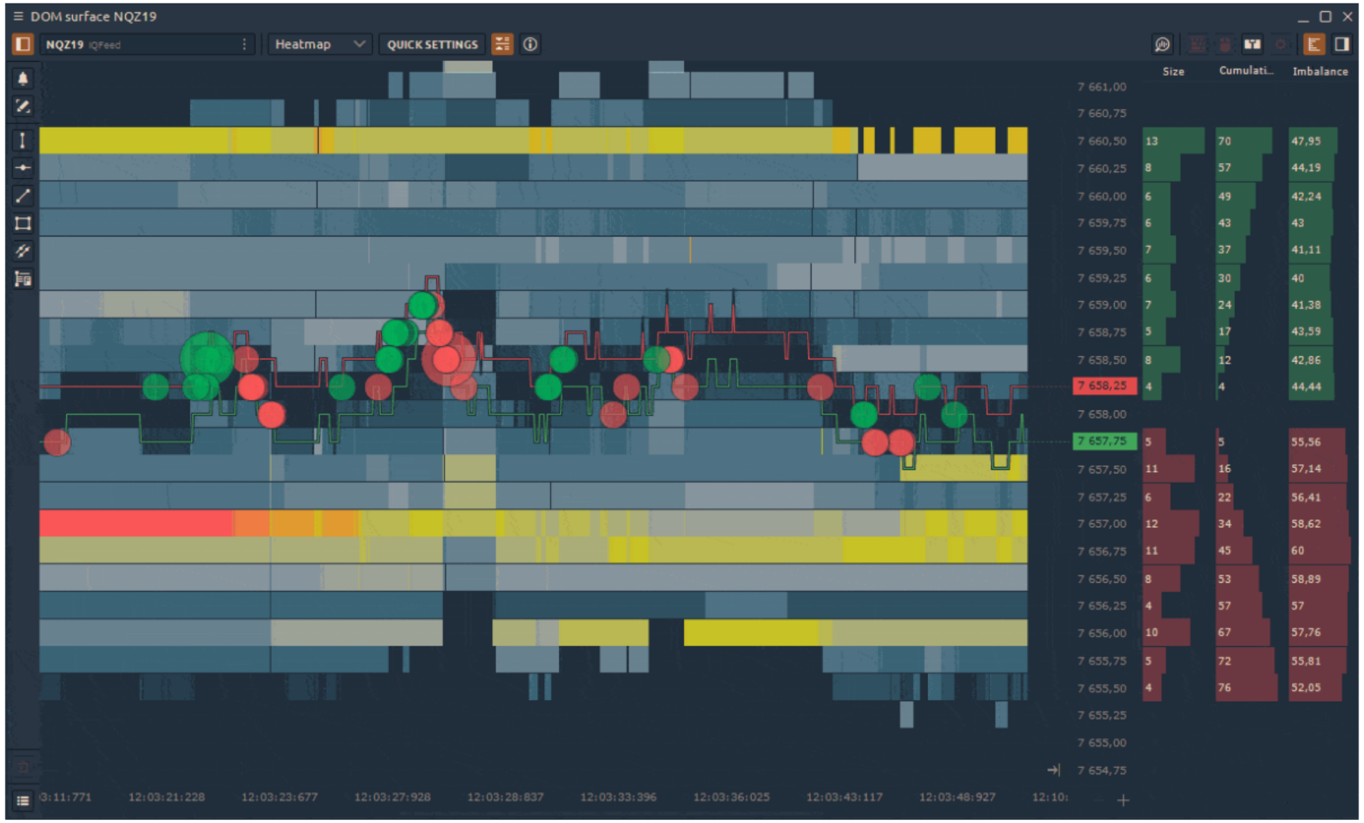
Two kinds of market microstructures

- Large ticksize (relative to price)
- Small ticksize (relative to price)

Large ticksize

- Futures
- Low priced stocks
- Liquid etfs

Nasdaq futures

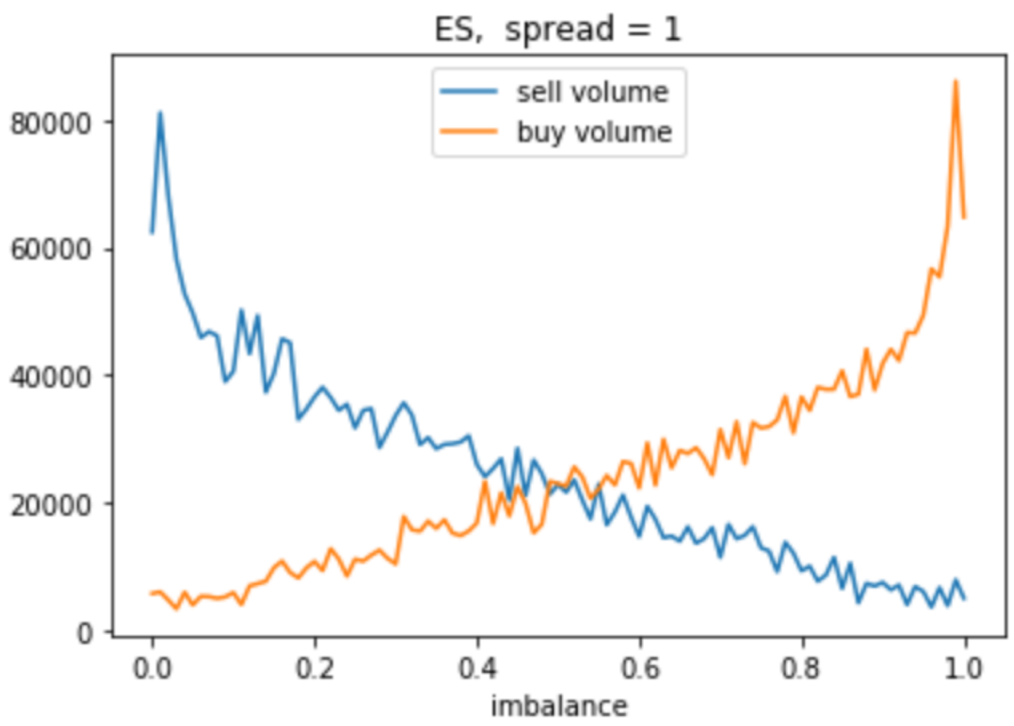


Imbalance is the least well kept secret in HFT

- $I_t = \frac{Q_t^b}{Q_t^b + Q_t^a}$ is related to the probability of the price moving up
- When imbalance is close to 1, traders are more likely to want to buy
- When imbalance is close to 0, traders are more likely to want to sell
- This is empirically true

In [9]:

```
plot_trade_imbalance(df)
```



The Microprice

Define

$$P_t^{micro} = \lim_{i \to \infty} P_t^i$$

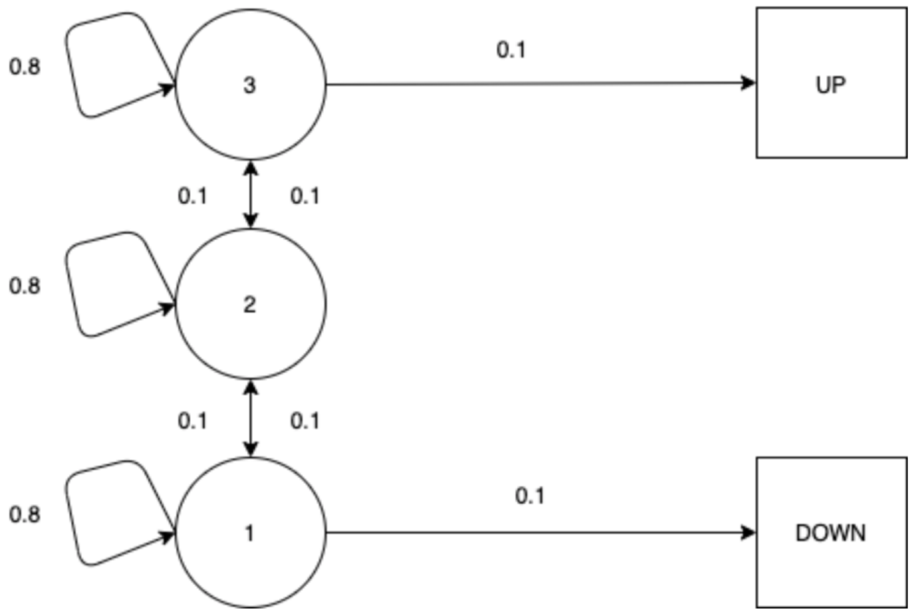
where

$$P_t^i = \mathbb{E}\left[M_{\tau_i} | \mathcal{F}_t\right]$$

$\tau_1, \ldots, \tau_n$ are (random) times when the mid-price M_t changes

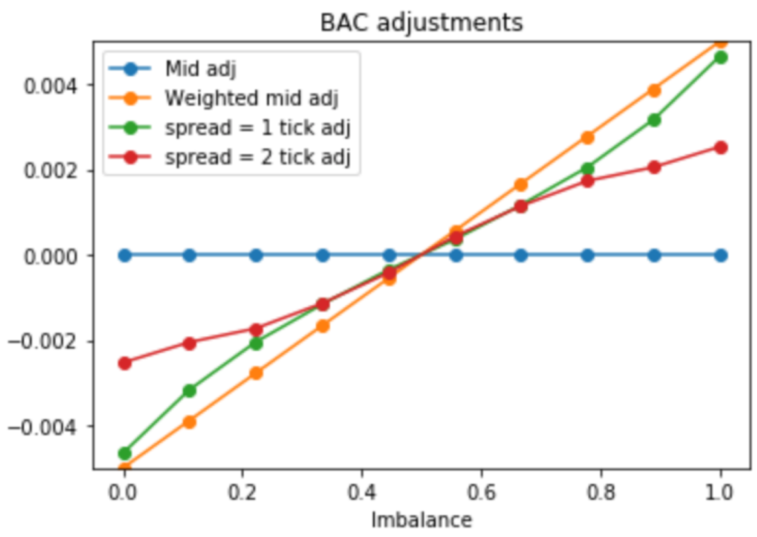
$\mathcal{F}_t$ is the information contained in the order book at time t, for example the spread and the imbalance

A toy example



Bank of America: a large tick asset

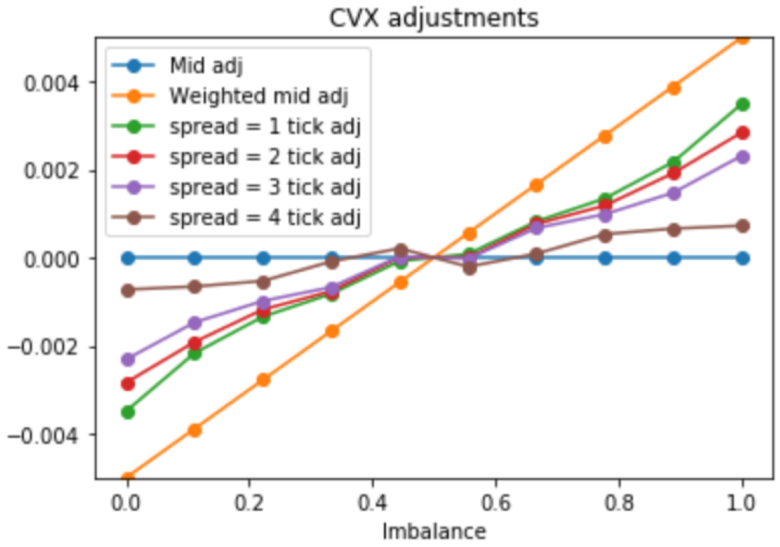
In [28]:
```
imb=np.linspace(0,1,n_imb)
G1,B,Q,Q2,R1,R2,K=estimate(T)
Gstar=plot_Gstar(ticker,G1,B,T,10)
```



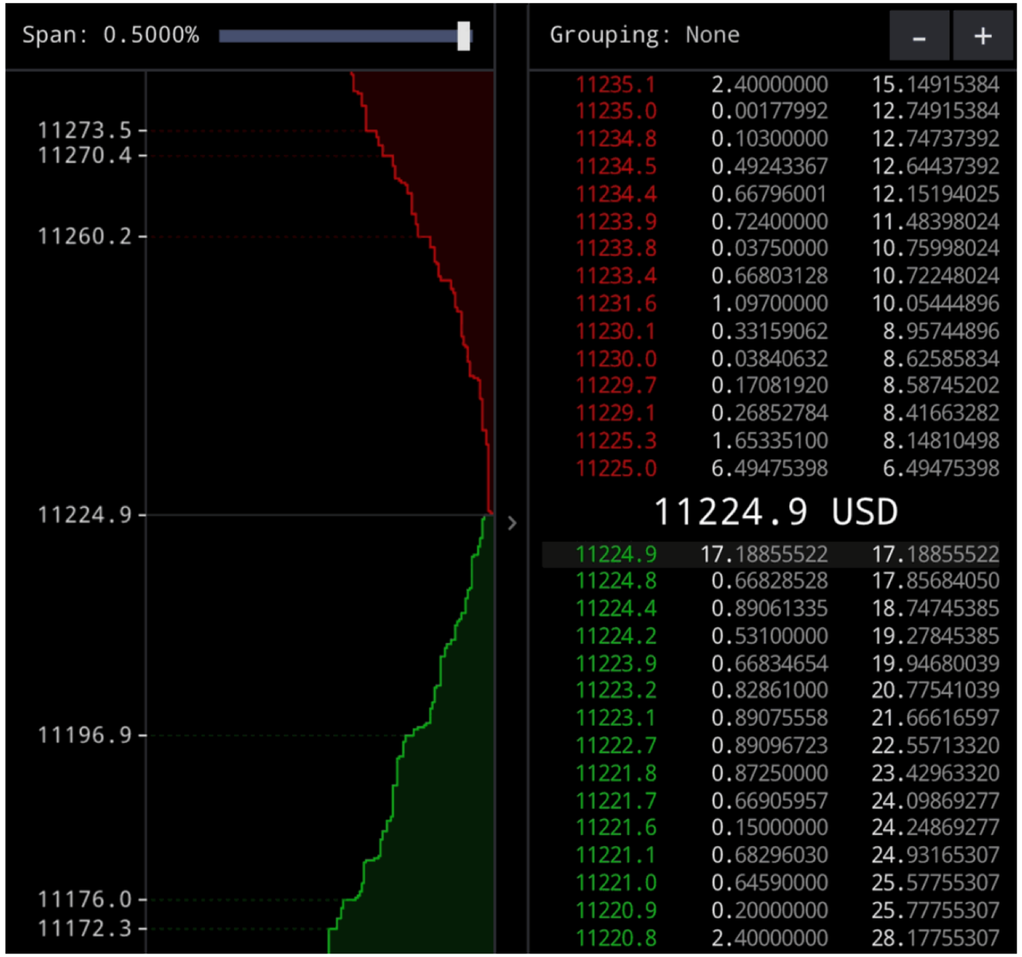
Chevron: a medium tick asset

In [30]:

```
Gstar=plot_Gstar(ticker,G1,B,T,10)
```



Bitcoin is a small tick asset



VAMP:

Volume adjusted mid price

$$P_b^{VWAP} = \frac{\sum_{i=1}^{n} P_b^i Q_b^i}{Q}$$

$$P_a^{VWAP} = \frac{\sum_{i=1}^{n} P_a^i Q_a^i}{Q}$$

$$VAMP = \frac{P_b^{VWAP} + P_a^{VWAP}}{2}$$

Covario Master thesis (2022)

Backtesting a simple market maker

Formulas in Hummingbot (2022)

$$r(s, q, t, \sigma) = s - q\gamma\sigma$$

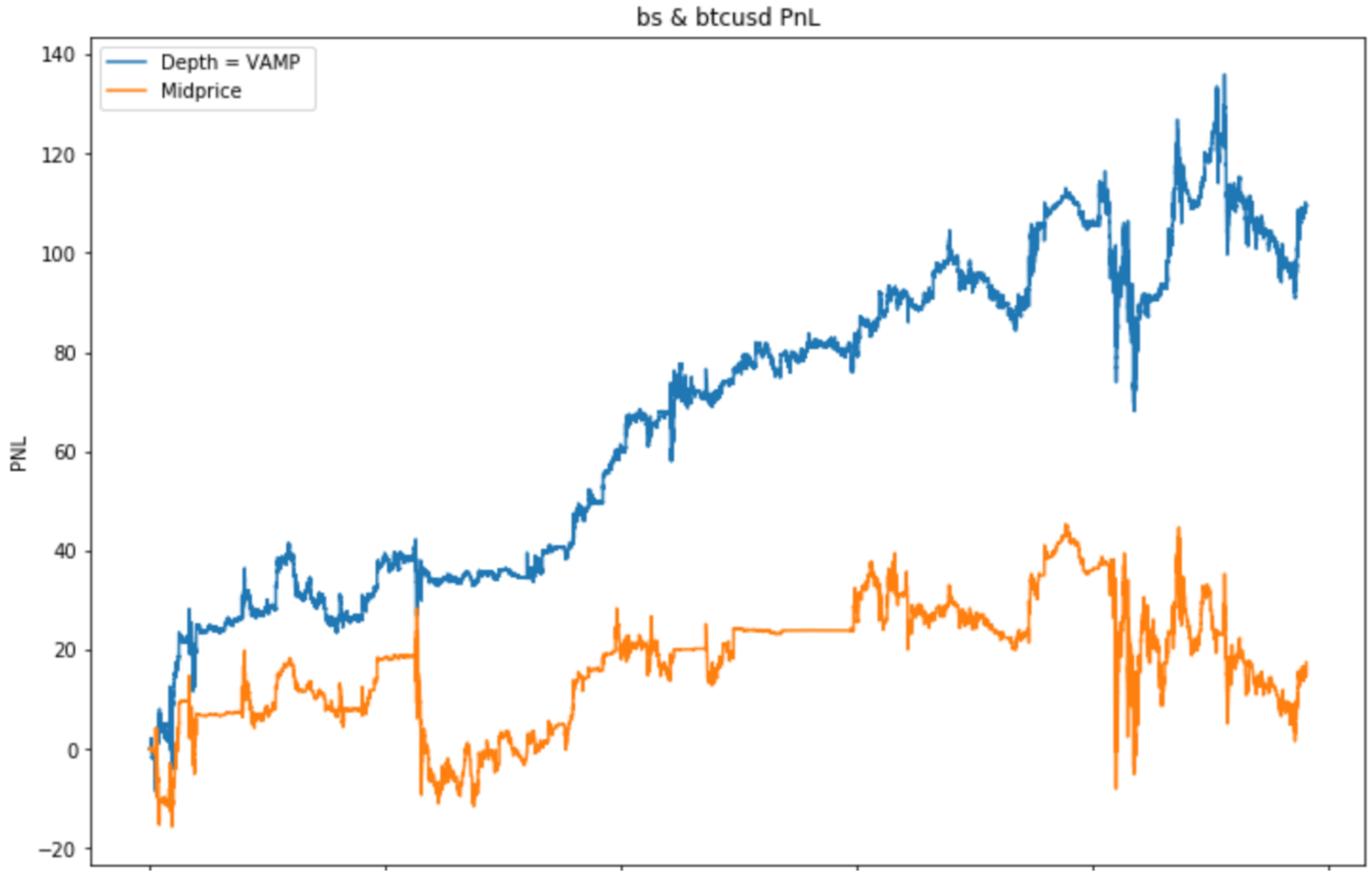
$$\delta^a + \delta^b = \gamma\sigma + \frac{2}{\gamma} ln(1 + \frac{\gamma}{\kappa})$$

Formulas in this simplified backtest

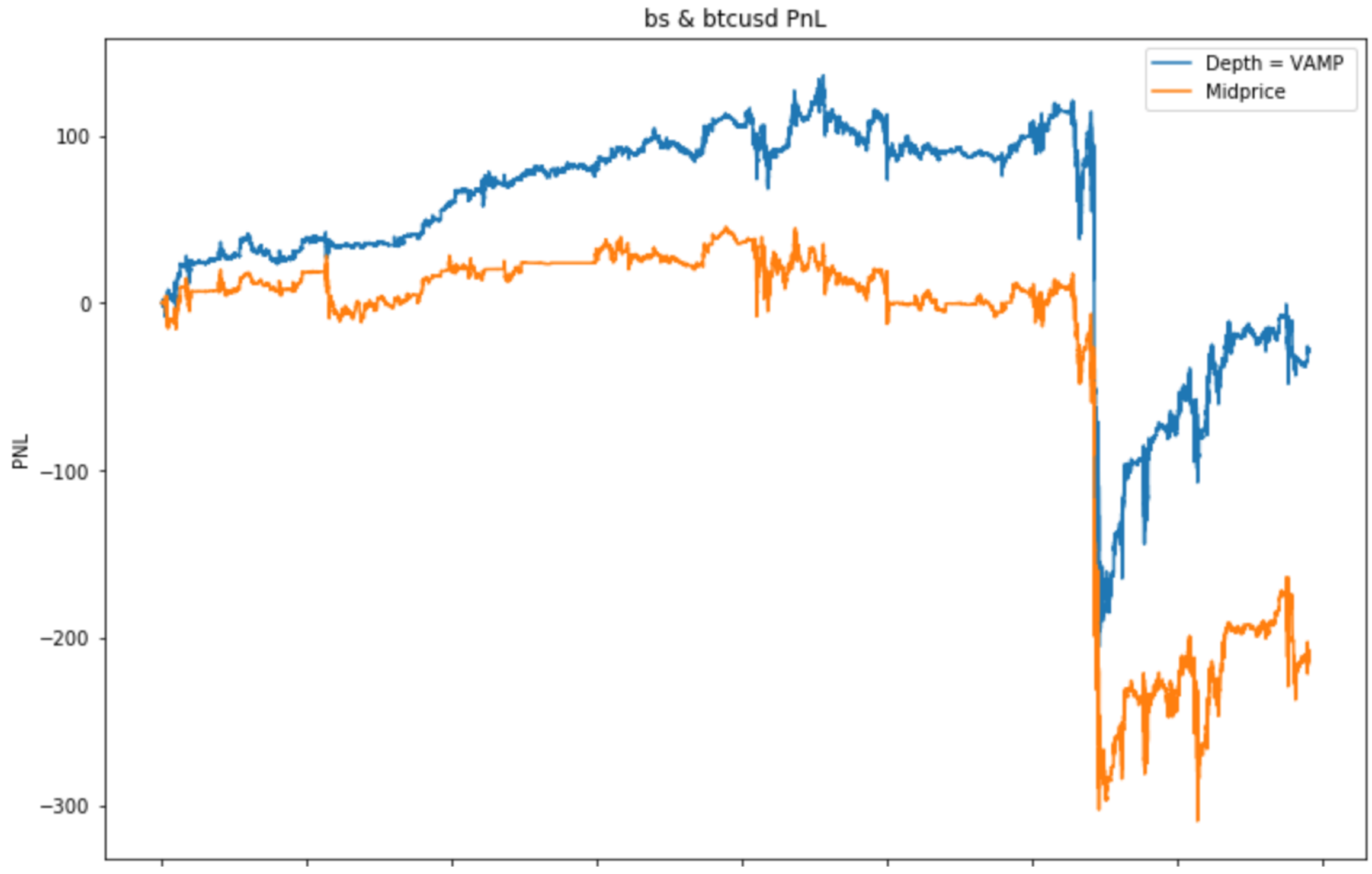
$$r(s, q, t, \sigma) = s - \beta q$$

$$\delta^a + \delta^b = k$$

Goal: Compare s=MIDPRICE to s=VAMP



bs & btcusd PnL
Depth = VAMP
Midprice
PNL
140
120
100
80
60
40
20
0
−20


bs & btcusd PnL
Depth = VAMP
Midprice
PNL
100
0
−100
−200
−300

Market Making is not easy!