Experiment 1: Primer

Henry Ashton

Transfer Viva

Jan 2020

Experiment 1: Spoofing Emergence

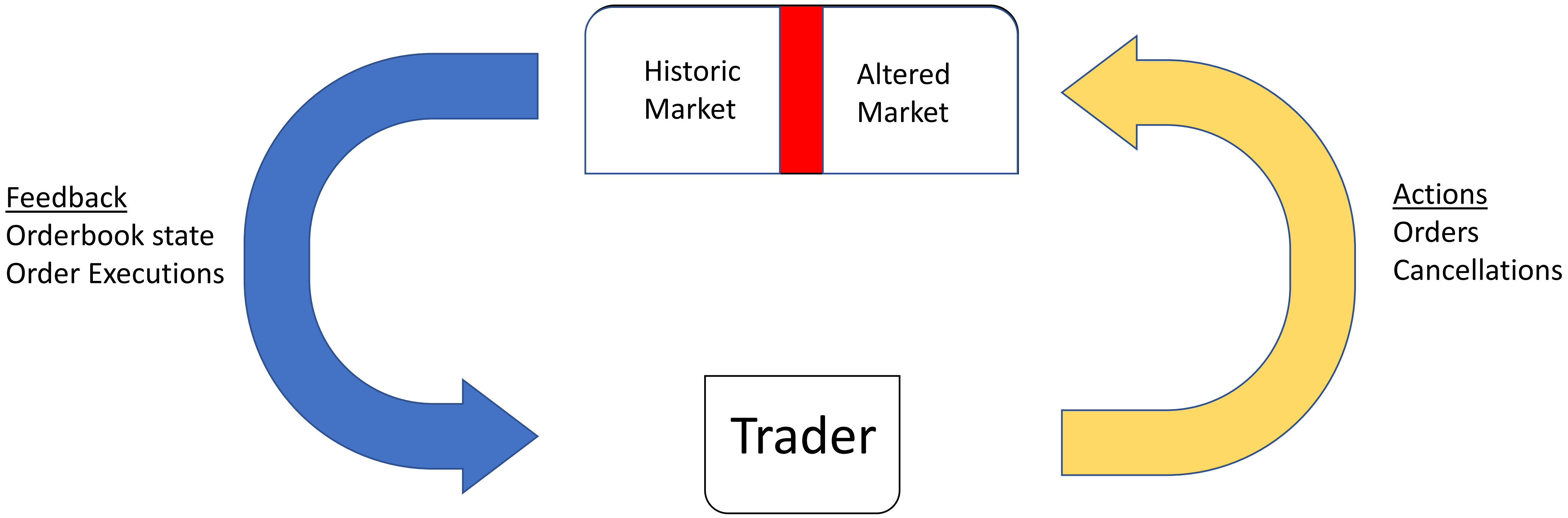
- **Research Question**: Will a self taught trading algorithm learn a market manipulative strategy?
- Presentation Structure:
  1. Method Motivation
  2. Method Description
  3. Results

Emergence of Market Manipulation

- A manipulating trading strategy will seek to control the state of the market by choosing a sequence of actions (orders) with which to influence it.
- Equivalently, the trader learns the market impact of their actions (orders) and optimizes their strategy to take advantage of this.

Problem 1: Data

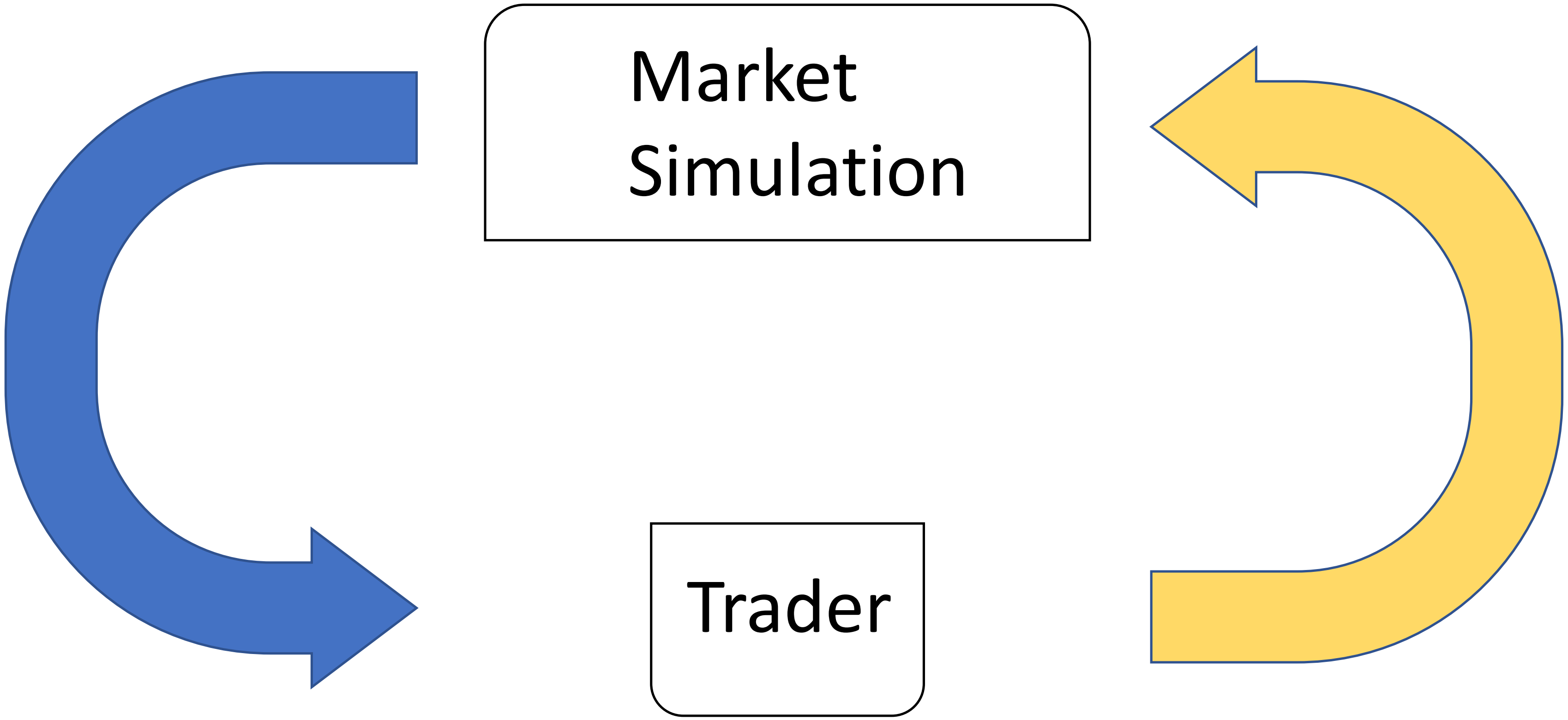
- Approach 1.1: Use historic data
- Problem 1.1: There is no feedback dependent on the trader’s actions; there can be no market manipulation



Problem 1: Data

- Approach 1.2: Use simulated data
- Problem 2: How do you simulate a market such that it responds to the actions of the trader?

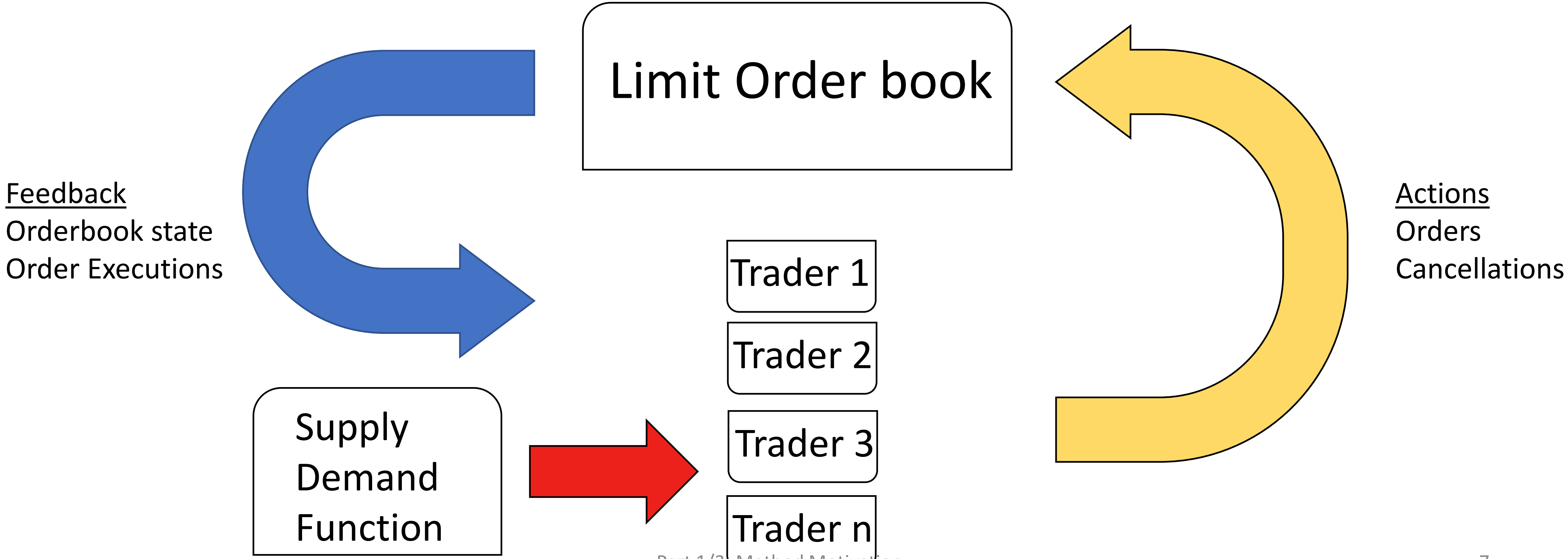
<u>Feedback</u>
Orderbook state
Order Executions



<u>Actions</u>
Orders
Cancellations

Problem 2: How do you simulate a market?

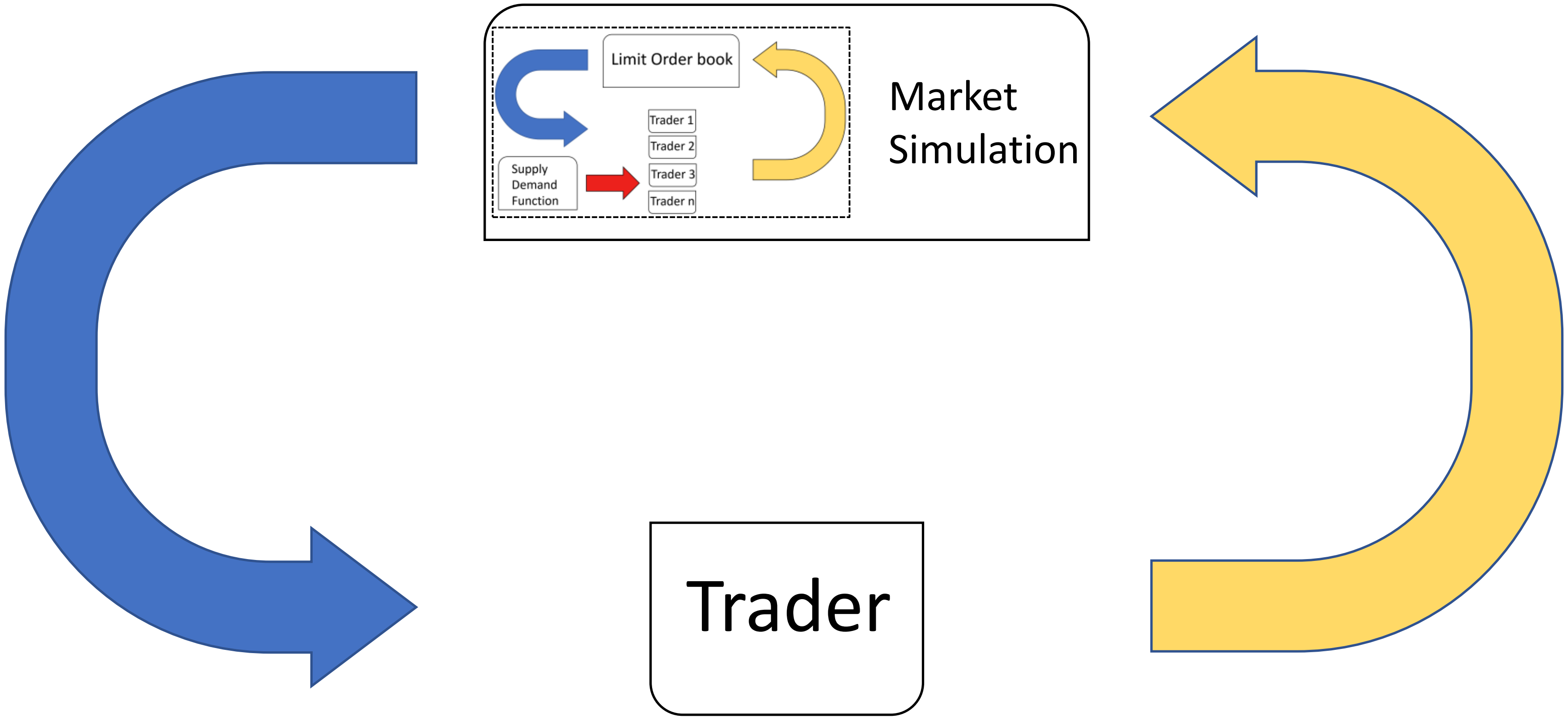
- Approach 2.1: Multi Agent Simulation
- Multiple 'Zero intelligence' traders driven by a supply and demand function interact with a working limit order book to drive the dynamic and response of a market simulator.
- Market dynamics and response are a function of the market mechanism ensuring some realism.



Problem 2: How do you simulate a market?

- Approach 2.1: A Multi Agent Simulation is the simulator
- For any learning Trader interacting with this market simulation, their actions will have feedback.

Feedback
Orderbook state
Order Executions



Actions
Orders
Cancellations

Reinforcement learning fits problem well

- Agent chooses actions which allow it to *interact* with an environment
- The Environment changes in response to those actions.
- The Agent receives feedback through the state information of environment and a reward
- The Agent typically learns to choose actions to maximise their cumulative reward over time

Experiment 1: Spoofing Emergence

- (*Simulation*) In a fully functioning limit order book market, …
- (*Setting*): … populated by zero intelligence traders, …
- (Supply and demand) … subject to a supply and demand regime, …
- (*RL setup*): … will a single trader, learning through Reinforcement learning, …
- (*RL method*): … learn a strategy where they are able to send orders to the exchange, …
- (*Interpretation*) … which is market manipulative to their advantage?

Simulation (Part 1 of 2)

- BUCLSE is a self developed minimal LOB simulator based on BSE
- One single tradeable asset.
- Each session is limited to 6000 discrete periods.
- During each period, one trader is selected at random to submit or cancel an order/orders to the exchange.
- After submission, any resulting trades are processed and traders informed of executions and order amendments.
- The new state of the order book is distributed to all traders

Setting

- Four types of ‘Zero intelligence’ (ZI) traders. 10 of each.
  - Informed Traders – Submit orders subject to a private signal of ‘fundamental’ asset value, knowledge of fundamental price process and a prior distribution.
  - Heuristic Belief Traders – Submit orders as informed traders but maximise expected surplus subject to estimate of probability of order being filled as a function of order price.
  - Momentum traders – Submit orders using average new order price with some memory.
  - Imbalance Traders – Choose to submit orders according to Orderflow imbalance statistic (Cont et al, 2013) with some memory.
- Inventory is limited in range for all traders [-5,5].
- Agent’s preference over inventory adjusts submitted order prices to deter larger inventories.
- Traders can have at most one bid and one ask order at any time.

Supply and demand regime

- Multi agent market simulations require a supply and demand function to drive the behaviour of the ZI agents which in turn drives the LOB.
- BUCLSE is suitable for a variety of different approaches.
- Experiment 1 uses the idea of a fundamental value sequence.
- This a random walk with mean reversion: $k\bar{r} + (1-k)r_{t-1} + \varepsilon_t = r_t$
- When asked to submit an order, the HBL trader receives a noisy signal about the fundamental value to update prior about future price.
- Supply and Demand are implicitly defined with this setup.
- Net holdings of the single asset are 0, net cash holdings are 0.
- Alternative would be to define supply and demand curve and assign orders to ZI traders as per original BSE specification.

RL Setup

- Task: RL Agent begins with inventory 1 and is asked to submit order each simulation period.
- **Termination**: Episode ends when inventory changes or when 100 periods have elapsed or when unrealised loss greater than 2.
- **Reward Function**: weighted change in unrealised profit and realised profit on trade execution. Penalty for acquiring additional inventory.
- **Actions**: Do nothing; Cancel orders; Submit bid order at best, quantity 1; Submit bid order at best, quantity 5; Sell inventory at best bid.
- **State space**:
  - **Distance** (total change in best bid from beginning), **Inventory**, **#Orders out**, **Bid Ask spread**, **Change Bid**, **Change Ask**, **Position in LOB** (0=front), **Orderflow imbalance**, **Time**
  - State space is scaled and clipped to simplify further.
- When episode ends, simulation environment is reset by allowing other traders to trade for 100 periods before RL Agent re-enters market.
- When time is up on trading session (6000 periods), a new one is generated, underlying price sequence is regenerated and traders are created afresh.

RL Setup – Episode Termination Criteria

- **A training episode ends if**:
  1. 100 periods have elapsed.
  2. Inventory is 0 (implying a sale)
  3. Inventory is greater than 1 (implying a buy)
  4. Unrealised loss is greater than some cutoff (2)

RL Setup – State space

- **State space**:
  1. **Distance** (total change in best bid from beginning)
  2. **Inventory**
  3. **#Orders out**
  4. **Bid Ask spread**
  5. **Change Bid**
  6. **Change Ask**
  7. **Position in LOB** (0=front)
  8. **Orderflow imbalance**,
  9. **Time remaining**

State space is scaled and clipped to simplify further.

RL Setup – Action space

- **Action space**:
  1. Do nothing
  2. Cancel orders
  3. Submit bid order at best, quantity 1 (naughty)
  4. Submit bid order at best, quantity 5 (naughty)
  5. Sell inventory at best bid.

- Only motivation for submitting bids is to increase best bid price because acquiring more inventory results in episode ending and penalty.

Simulation (Part 2 of 2 – incorporates RL trader)

- BUCLSE is a self developed minimal LOB simulator based on BSE
- One single tradeable asset.
- Each session is limited to 6000 discrete periods.
- During each period:
  - RL Agent is invited to submit order, resulting trades processed, traders updated with new state of order book
  - One trader is selected at random, they receive noisy signal about fundamental value, then submit cancels or bid or ask to the exchange.
  - After submission, any resulting trades are processed and traders informed of executions and order amendments.
  - The new state of the order book is distributed to all traders
- RL Agent experiences 10,000 sessions.

RL Method

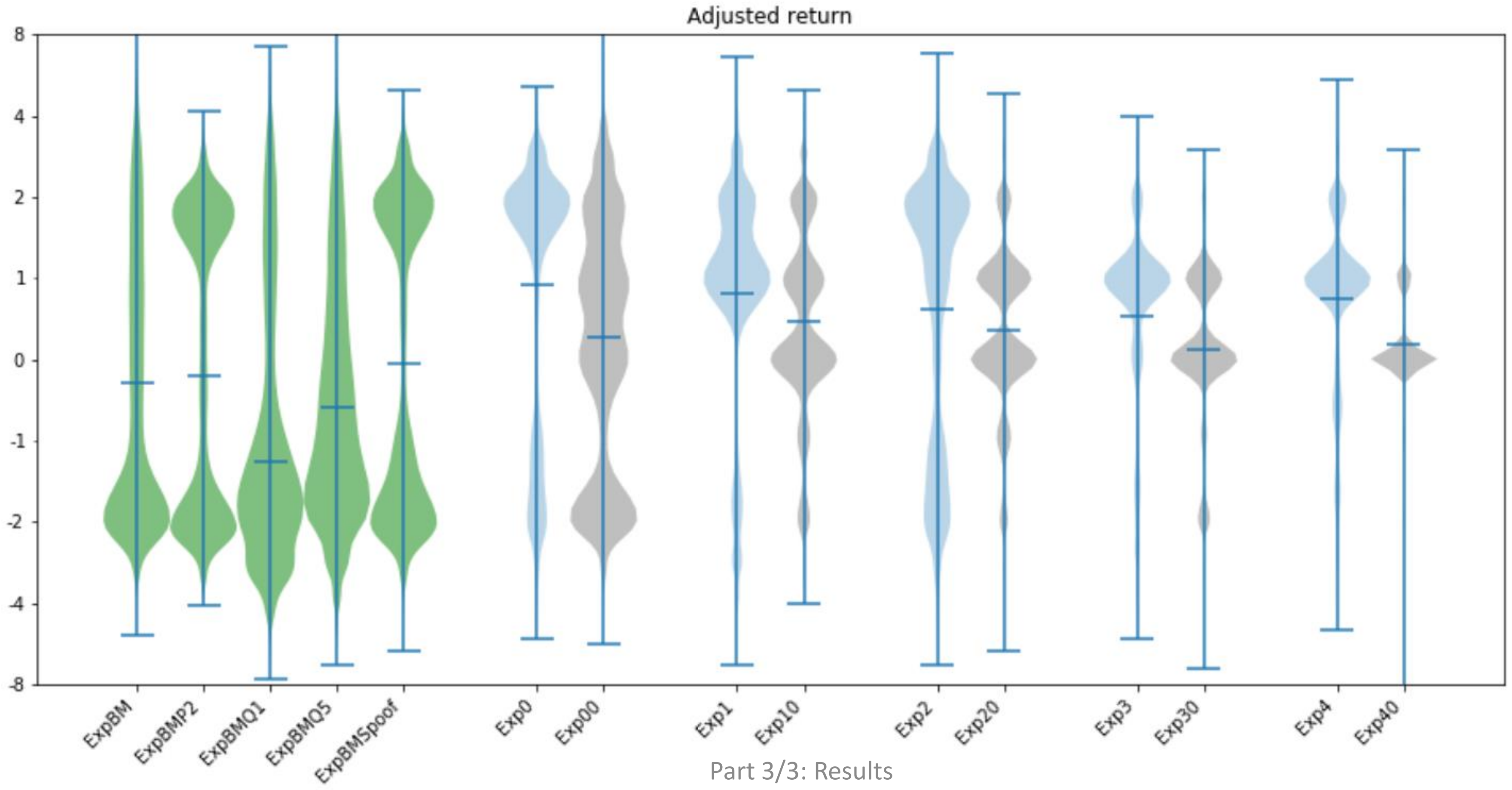
- RL Agent learns through Q learning sample (one step temporal diff)
- Attempts to learn a neural network which is a parameterised estimator of the optimal Q value for any state, action pair
- Also try Dyna-Q learning where agent additionally fits a model to predict change in state given an input action and state.
- Models tried in Dyna-Q setup: Straightforward empirical estimator and Conditional Variational Autoencoder.
- Exploration methods include epsilon greedy and UCB
- Full tabular learning also attempted with full backups.
- Candidate optimal policies are evaluated in separate test loop.
- Best policy after 10,000 training episodes is tested over 5,000 test episodes.

Interpretation

- RL agents are trained with full action space and limited action space (no bid order submission actions) for each setup.
  - Difference in performance indicates how useful submitting limit orders is to RL Agent.
- Analysis of why trading episode ends and choice of actions indicates how successful RL Agent is at balancing order submission with avoiding trade execution.
- Training a tree classifier to approximate neural network, allows intuitive interpretation of found optimal strategies.

Results

- In all experiment settings, RL agent is able to use enlarged action space to improve their profitability. (Exp's ending '0' denote restricted action space)



Results

- Upper chart shows distribution of strategy returns for full action space (LHS) and restricted action space (RHS)
- Lower chart shows distribution of order flow imbalance during experiments.

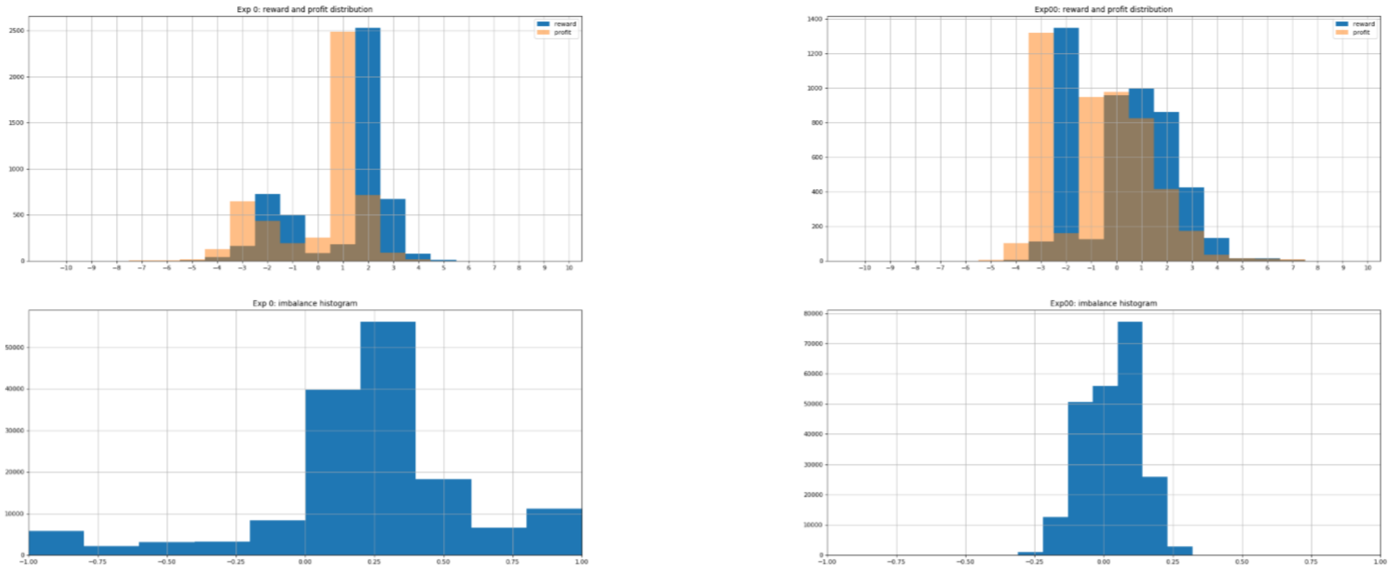


Figure 4.5: Upper charts shows distribution of returns associated with the best policy found in Experiments 0 and 00. Lower chart shows distribution of the order book imbalance metric during trading. The value shown is scaled by a factor of 0.05 and clipped to be in the range in [-1,1] hence the peaks at either end. Observe narrow distribution of imbalance in Exp00 on the right panel where the trader has no ability to affect it.

Results

- Tree classifier fit to best strategy with accuracy 90%, where importance of each class is equally weighted.
- Decision sequence indicative of spoofing strategy