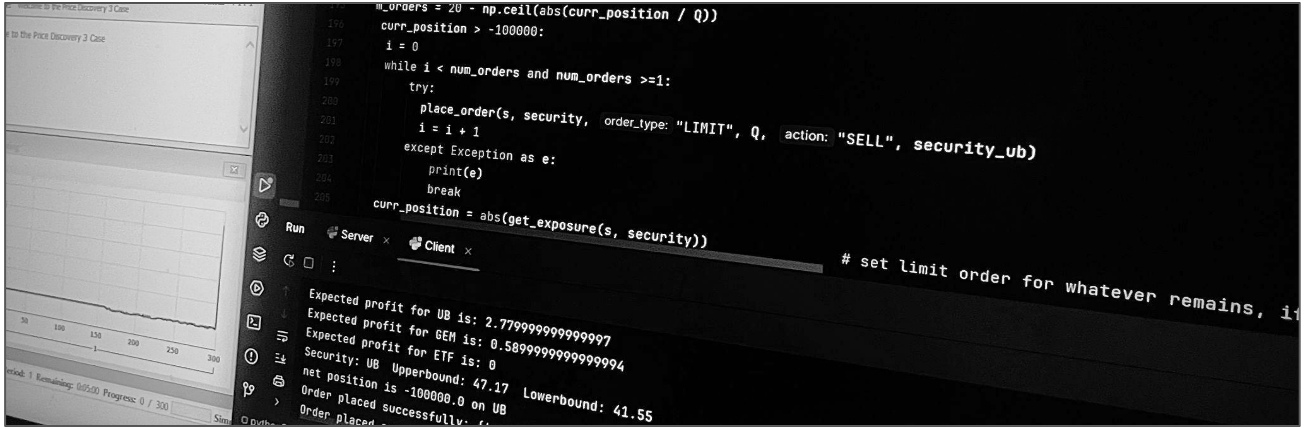
Leveraging Network Programming for Long/Short Algorithmic Trading



```
m_orders = 20 - np.ceil(abs(curr_position / Q))
curr_position > -100000:
 i = 0
 while i < num_orders and num_orders >=1:
     try:
       place_order(s, security,  order_type: "LIMIT", Q,  action: "SELL", security_ub)
       i = i + 1
     except Exception as e:
         print(e)
         break
curr_position = abs(get_exposure(s, security))
                                                   # set limit order for whatever remains, i
```

Welcome to the Price Discovery 3 Case
e to the Price Discovery 3 Case
50
100
150
200
250
300
Period: 1 Remaining: 0:05:00 Progress: 0 / 300
Run
Server
Client

```
Expected profit for UB is: 2.7799999999999997
Expected profit for GEM is: 0.5899999999999994
Expected profit for ETF is: 0
Security: UB  Upperbound: 47.17  Lowerbound: 41.55
net position is -100000.0 on UB
Order placed successfully:
Order placed
```

Group 7
Riccardo G. Corradi (4015 7648)
Daniel Perez Domouso (4015 6560)
SETO Fu Yuet (5718 6372)
YUNG Wing Ho (5675 1460)
Elijah J. Clark (4015 7022)

Introduction

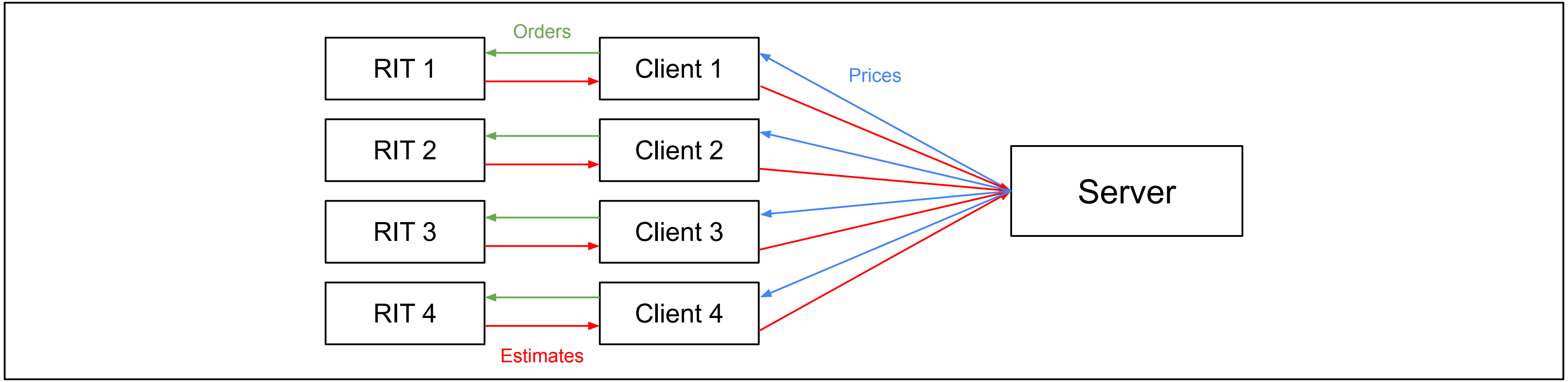
Our strategy involves four **clients** and one **server** communicating to maximize profit opportunities. In this presentation, we will break down:

- Network communication and bounds calculations
- Client trading logic; initial order placement & range optimization
- Why rebalance positions and how our rebalancing procedure works
- Simulation results and why this is the optimal strategy

Understanding the Power of a Network Structure

Combining information across traders yields an **informative advantage**:

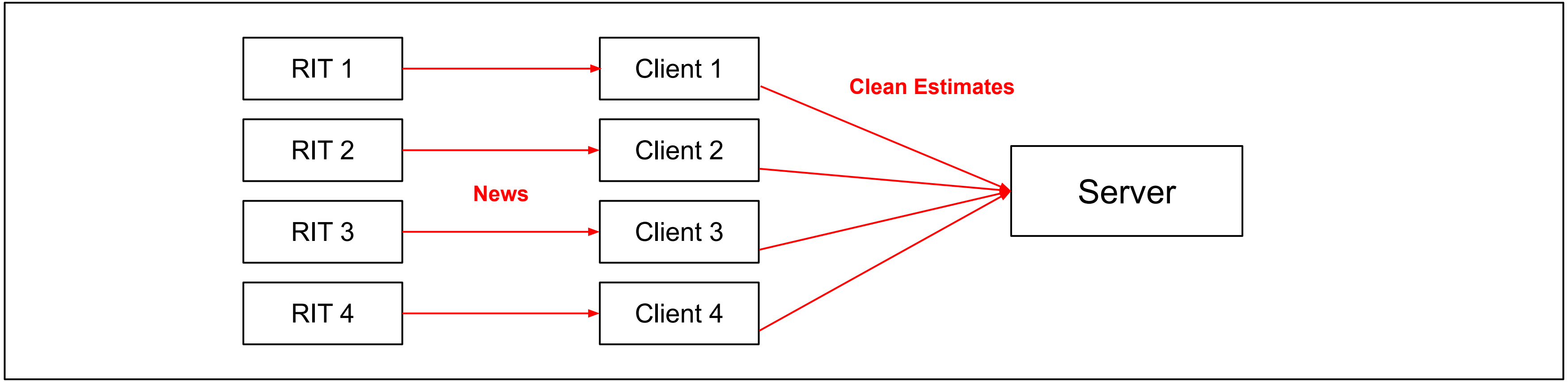
- Combining news from 4 traders gets more accurate information before others
- Trading on accurate estimates is also a loss-free strategy
- We achieve this through a scheme of 4 client programs and 1 server program



Information Flows from Interfaces to Server

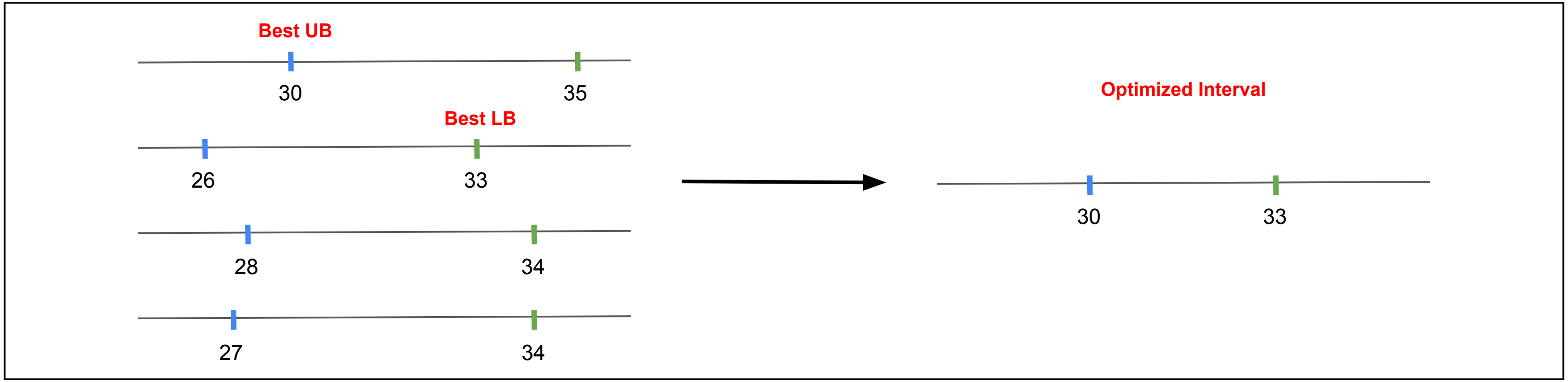
When a new news item appears in the RIT Interface:

- Clients fetch the time and the associated estimate from the news item’s body
- Clients send estimates, time, and ticker to the server
- The server combines information to compute tighter price ranges for each security



Leveraging Information for an Optimized Interval

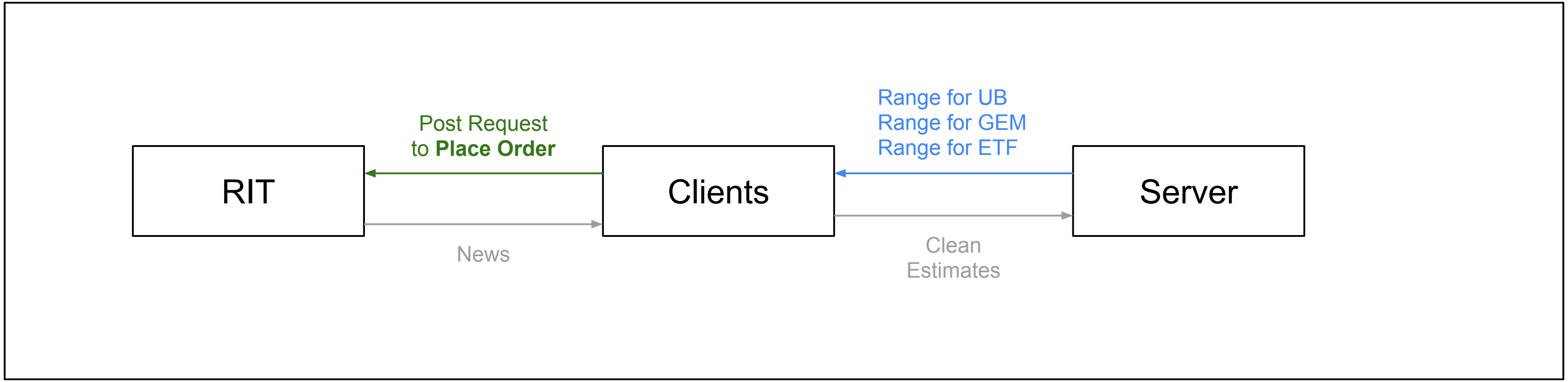
- Each trader receives an estimate, resulting in individual upper and lower bounds.
- The server aggregates individual estimates to produce an **optimized interval**.
- The final interval determine the trading opportunities.



From the Server’s Calculations to Actual Orders

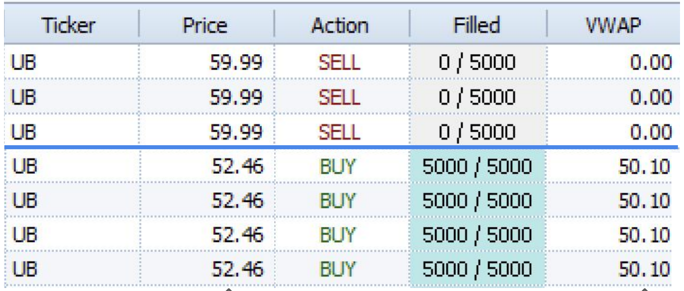
Once the server has the ranges:

- The server sends to each client the ranges for each security
- Clients detect profit-maximising arbitrage opportunity
- Clients place limit orders at the best bounds, which likely get filled immediately



Order Placements

- Profitable orders are placed first, getting filled instantly at the best bid/ask
- Limit orders are then placed on the opposite bound
- This ensures we are always taking advantage of opportunities on either side
- Limit prices are continuously updated to reflect new information



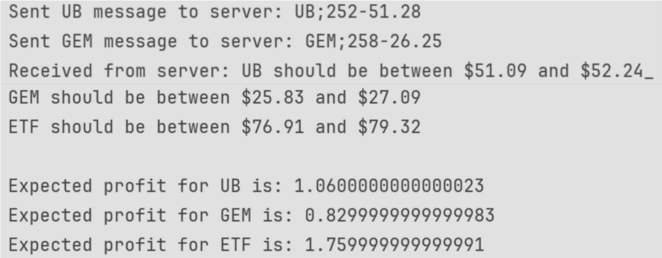
Ticker	Price	Action	Filled	VWAP
UB	59.99	SELL	0 / 5000	0.00
UB	59.99	SELL	0 / 5000	0.00
UB	59.99	SELL	0 / 5000	0.00
UB	52.46	BUY	5000 / 5000	50.10
UB	52.46	BUY	5000 / 5000	50.10
UB	52.46	BUY	5000 / 5000	50.10
UB	52.46	BUY	5000 / 5000	50.10

Bound Price

At least $2.36/sh. guaranteed profit

Filled Price

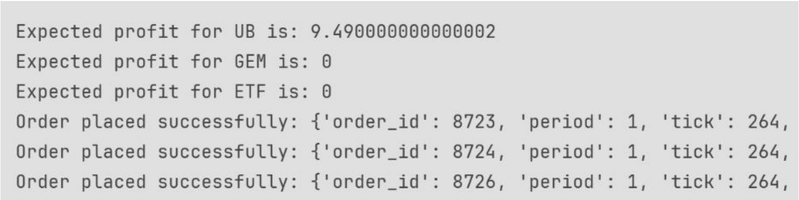
Communication



```
Sent UB message to server: UB;252-51.28
Sent GEM message to server: GEM;258-26.25
Received from server: UB should be between $51.09 and $52.24
GEM should be between $25.83 and $27.09
ETF should be between $76.91 and $79.32

Expected profit for UB is: 1.0600000000000023
Expected profit for GEM is: 0.8299999999999983
Expected profit for ETF is: 1.759999999999991
```

Order Placement



```
Expected profit for UB is: 9.490000000000002
Expected profit for GEM is: 0
Expected profit for ETF is: 0
Order placed successfully: {'order_id': 8723, 'period': 1, 'tick': 264,
Order placed successfully: {'order_id': 8724, 'period': 1, 'tick': 264,
Order placed successfully: {'order_id': 8726, 'period': 1, 'tick': 264,
```

Some Caveats to our Strategy & Solution for Further Profits

Current strategy has some **impediments to profit maximising**:

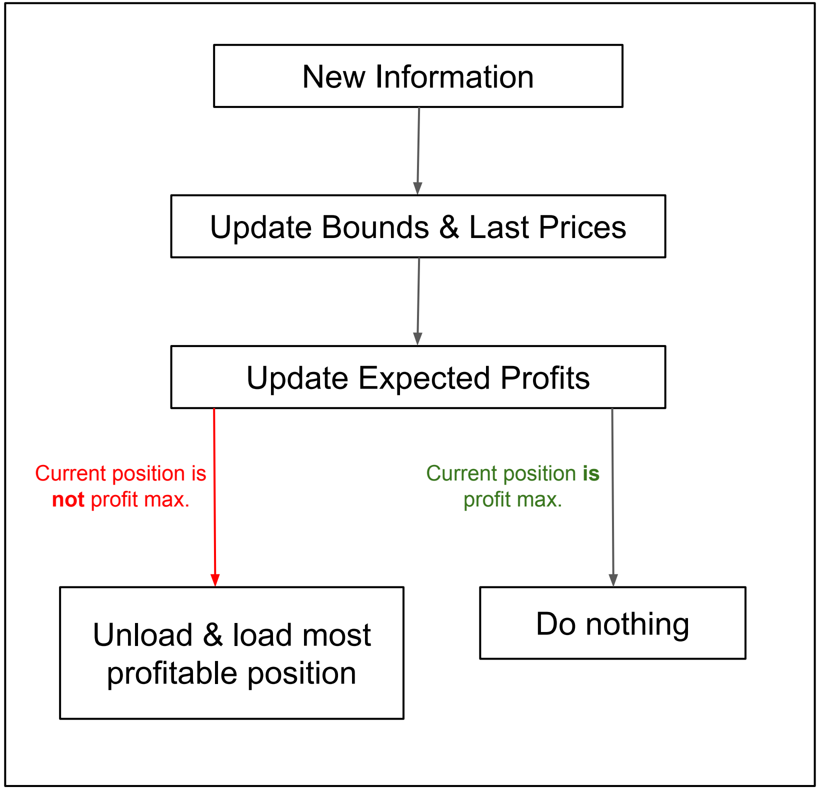
- Trade based on the current news only, not speculating on possible future better trading opportunities.
- New estimates may reveal the security being hold is no longer the profit maximizer.
- Price fluctuations may make one stock a better opportunity than another

Solution: **Rebalancing Protocol**

- Update security price bounds, and re-compute profits based on the updated information.

Even Further Profits: Position Rebalancing and its Mechanics

- Check if current holding security is the one that offers the highest profit, given the newest information.
- Check alternatives offering a higher profit than the expected one. Client will unload & build new more profitable position
- Significant increase in profits through rebalancing.

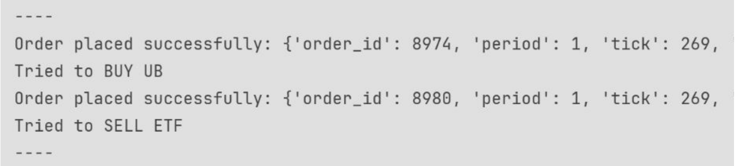

New Information
Update Bounds & Last Prices
Update Expected Profits
Current position is **not** profit max.
Current position **is** profit max.
Unload & load most profitable position
Do nothing

Rebalancing Mechanics, cont.

We also add a **tolerance** to our rebalancing protocol to offset market impact risk.

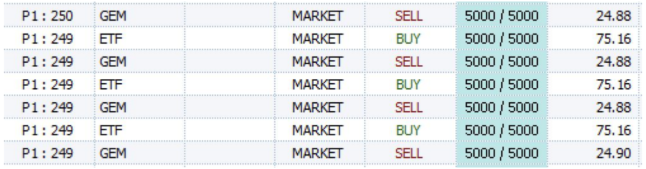
- *Tolerance* is the minimum extra profit per share a different security offers to be convinced that rebalancing ensures greater returns.
- This prevents repetitive rebalancing and adjusts for the market impact we will have when switching positions.
- Our tolerance for any rebalance involving GEM is higher, as GEM orders cause more price impact

e.g., code output



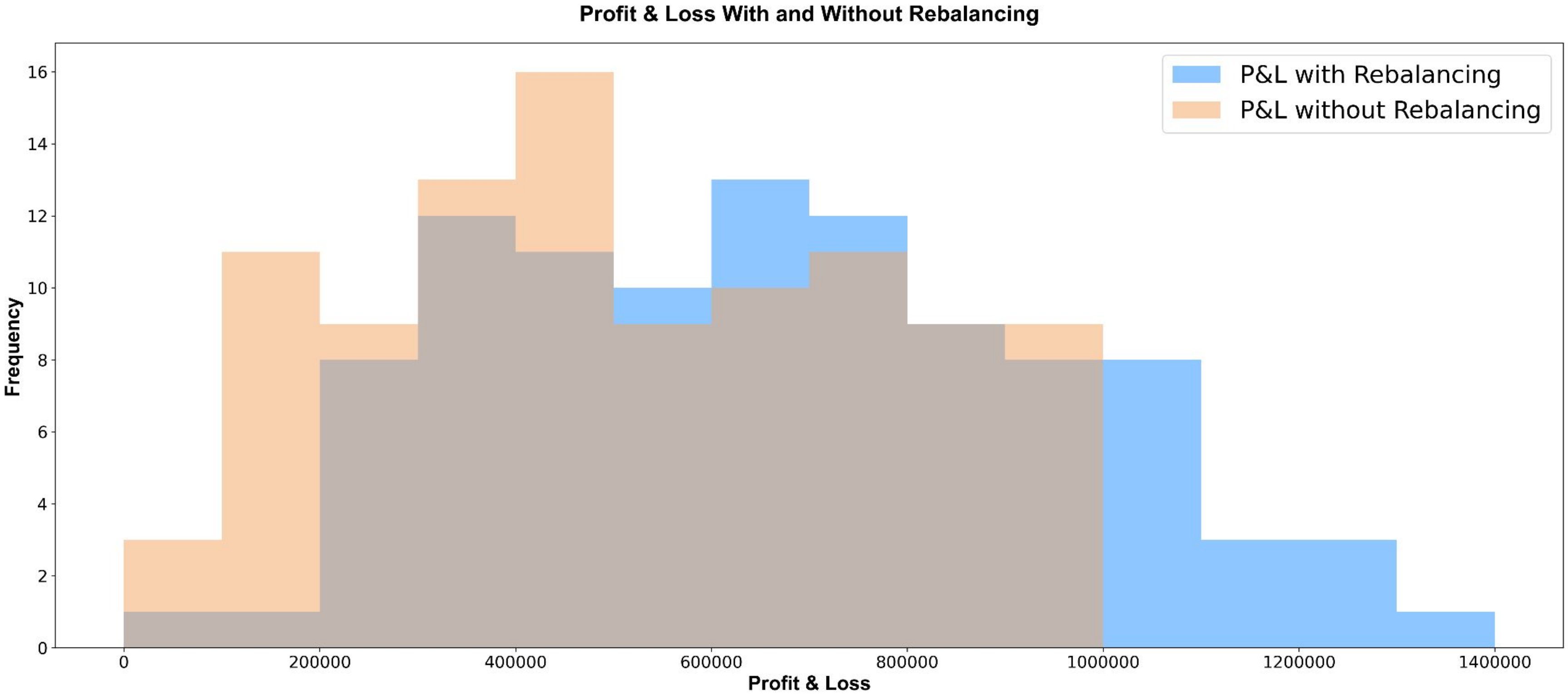
```
----
Order placed successfully: {'order_id': 8974, 'period': 1, 'tick': 269,
Tried to BUY UB
Order placed successfully: {'order_id': 8980, 'period': 1, 'tick': 269,
Tried to SELL ETF
----
```

e.g., RIT book

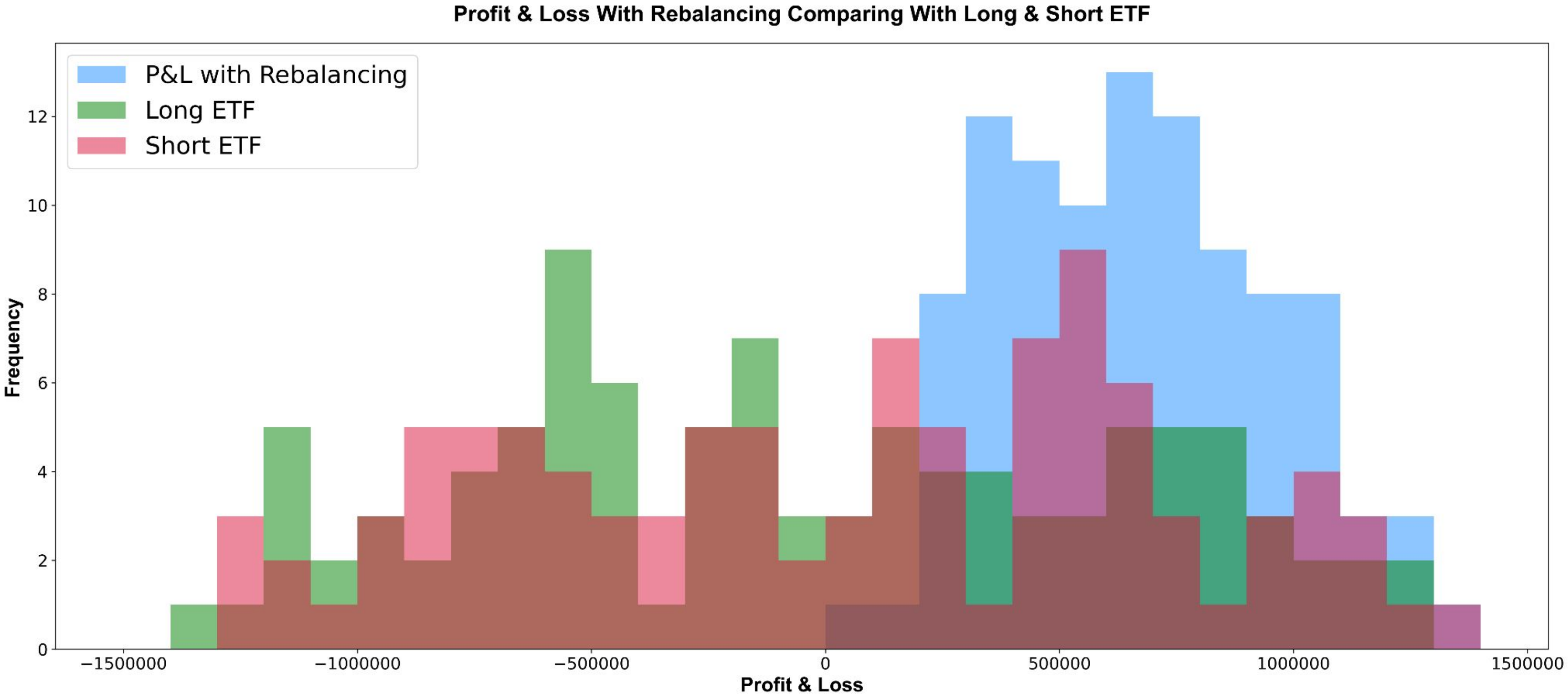


P1 : 250	GEM		MARKET	SELL	5000 / 5000	24.88
P1 : 249	ETF		MARKET	BUY	5000 / 5000	75.16
P1 : 249	GEM		MARKET	SELL	5000 / 5000	24.88
P1 : 249	ETF		MARKET	BUY	5000 / 5000	75.16
P1 : 249	GEM		MARKET	SELL	5000 / 5000	24.88
P1 : 249	ETF		MARKET	BUY	5000 / 5000	75.16
P1 : 249	GEM		MARKET	SELL	5000 / 5000	24.90

P&L With and Without Rebalancing

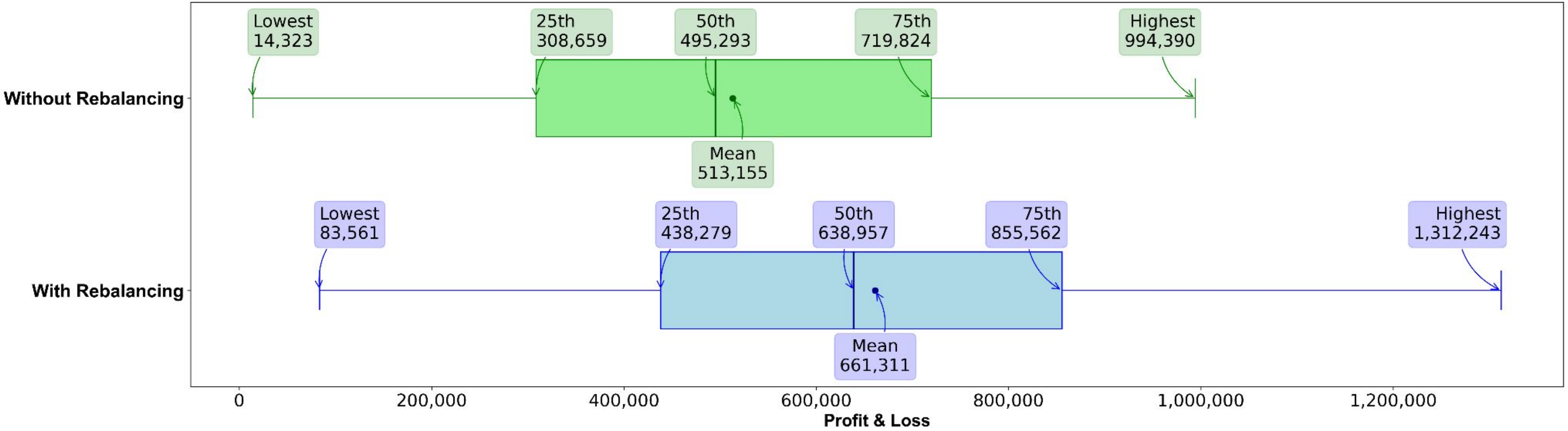


P&L With Rebalancing vs Long & Short ETF



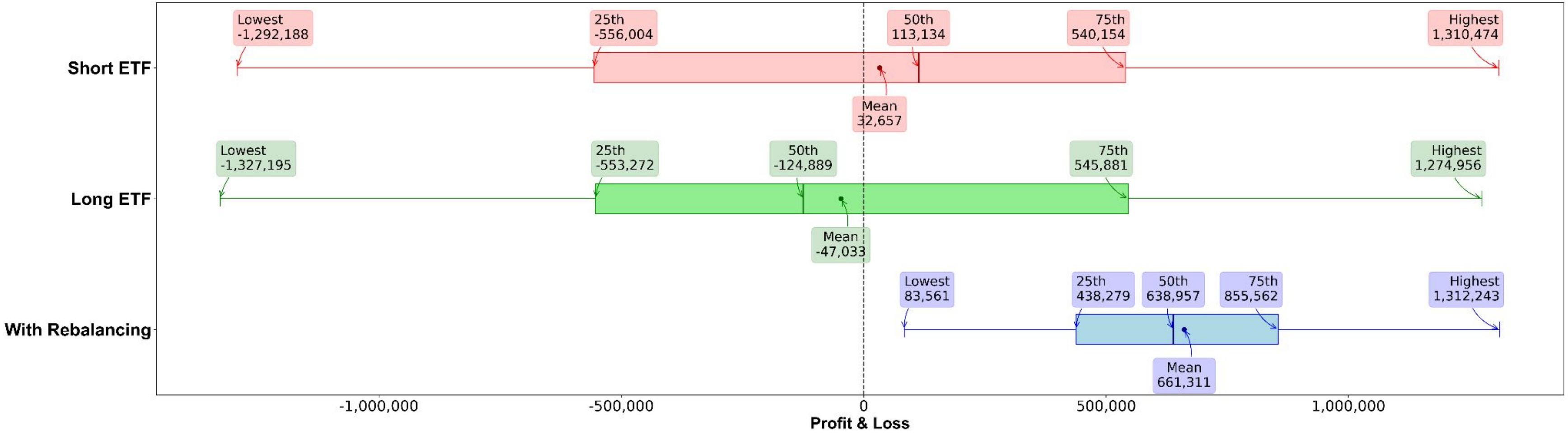
Significant Increase in Profits through Rebalancing

- Rebalancing yields an **incremental average profit of 150.000**
- **Positive** 99th percentile **Value at Risk (VaR)** equal to **159.285**



Rebalancing offers more profit relative to passive benchmarks

- Rebalancing yields a much higher mean profit compared to the other passive strategies
- The variability in return is stabilized with rebalancing showing more consistent performance and reduced downsides risk
- Rebalancing allows a tighter range of return, minimizing extreme losses while maintaining consistency across various market scenarios



Rebalancing offers more wins than losses relative to any benchmark.

P&L Difference between Strategies	When Rebalancing Strategy Outperforms	When Rebalancing Strategy Underperforms	Overall
Rebalancing vs Non-Rebalancing	183.090	(10.988)	148.156
Rebalancing vs Winning ETF	161.223	(35.014)	76.841

- Our strategy outperforms even a hypothetical 100% win-rate ETF trader.
- Before factoring in risk, our returns are already superior.
- **Risk-Adjusted Advantage**: When accounting for risk, our strategy dominates—offering near-zero downside.

Outperformance expressed in percentage terms

- "Outperformance" is better expressed in percentage terms.
- When the rebalancing strategy outperformed the winning ETF, it did so by **136.99%** of the ETF’s profits.
- When the winning ETF outperformed the rebalancing strategy, it did so by just **6.23%** of the rebalancing strategy profit

Absolute Avg. PnL% Difference between Strategies	When Rebalancing Strategy Outperforms	When Winning ETF Outperforms
Rebalancing vs Winning ETF	136,99%	6,23%

Assessing the Statistical Significance of Profit Differences

We show that, on average, the rebalancing strategy offers higher profits than the non-rebalancing and winning ETF strategies, and *the difference is statistically significant.*

- We first define the variables:

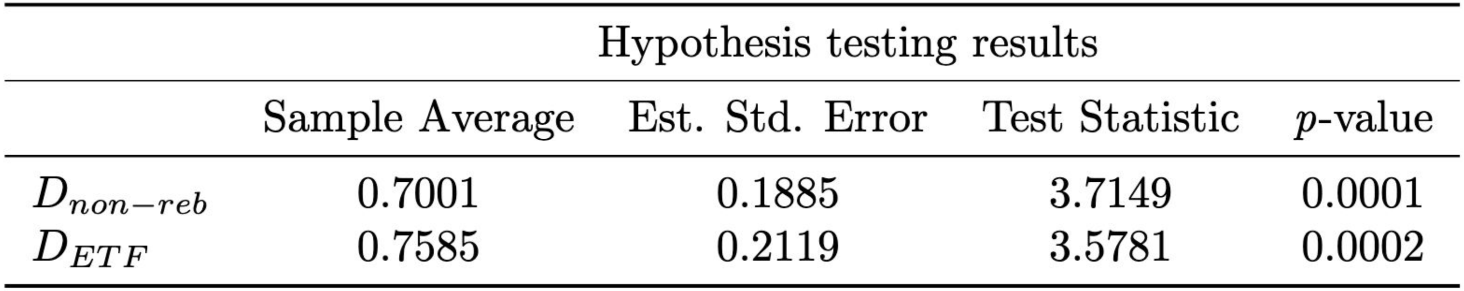
$$D_{non-reb} = \frac{\pi_{reb} - \pi_{non-reb}}{\pi_{non-reb}} \qquad D_{ETF} = \frac{\pi_{reb} - \pi_{ETF}}{\pi_{ETF}}$$

- We test the following hypotheses:

$$\begin{cases} H_0 : \mu(D_{non-reb}) \leq 0 \\ H_1 : \mu(D_{non-reb}) > 0 \end{cases} \qquad \begin{cases} H_0 : \mu(D_{ETF}) \leq 0 \\ H_1 : \mu(D_{ETF}) > 0 \end{cases}$$

Assessing the Statistical Significance of Profit Differences, cont.

We draw 100.000 samples of size 100 from the simulation data to **bootstrap standard errors** of the estimators. We refer to a T-distribution with 99 degrees of freedom to compute p-values. We used percentage differences in performance as a more conservative approach; nominal differences are quite large and easier to prove significant.



	Hypothesis testing results			
	Sample Average	Est. Std. Error	Test Statistic	p-value
$D_{non-reb}$	0.7001	0.1885	3.7149	0.0001
D_{ETF}	0.7585	0.2119	3.5781	0.0002

Conclusion

- **Innovative Strategy:** Leveraged client-server architecture for superior information aggregation and efficient execution.
- **Proven Performance:** Through testing, our rebalancing strategy outperforms benchmarks, provides near-zero downside risk, and is well-prepared for competitive simulations.
- **Adaptable & Robust:** Addressed liquidity, volatility, and multi-security exposure for consistent success in diverse market conditions.

Thank you very much for listening.

We will now leave a couple of minutes to any Q&A from you!