Мульти-агентная имитация процессов на биржах и маркетплейсах для оптимизации стратегий активного управления портфелями и борьбы с мошенничеством

Антон Колонин

akolonin@aigents.com

Facebook: akolonin

Telegram: akolonin

SingularityNET

AIGENTS

https://aigents.com

Novosibirsk State University

*THE REAL SCIENCE

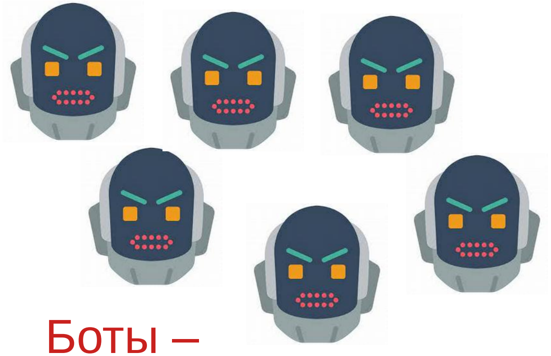
Какие проблемы решаем?

Каким образом устроить общество, чтобы максимизировать благополучие и безопасность порядочным гражданам в ущерб мошенникам (“на благо общества”)?

Как подобрать выигрышную стратегию на финансовом (валютном) рынке для себя лично, в ущерб всем остальным (“zero sum game”)?

Имитационное моделирование маркетплейса

Продавец -
мошенник



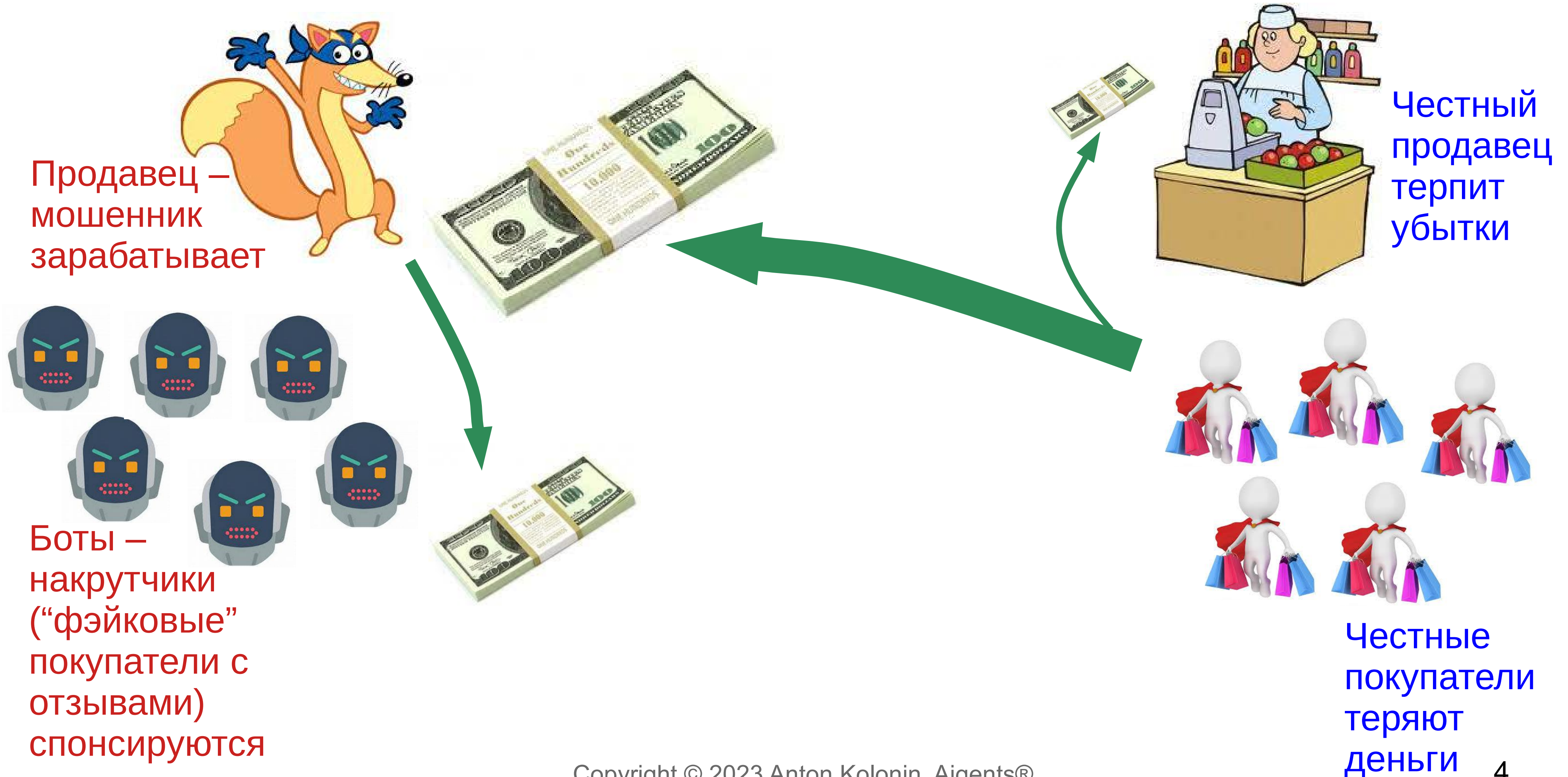
Честный
продавец

Боты –
накрутчики
(“фэйковые”
покупатели с
отзывами)

Каким образом устроить общество, чтобы максимизировать благополучие и безопасность порядочным гражданам в ущерб мошенникам (“на благо общества”)?

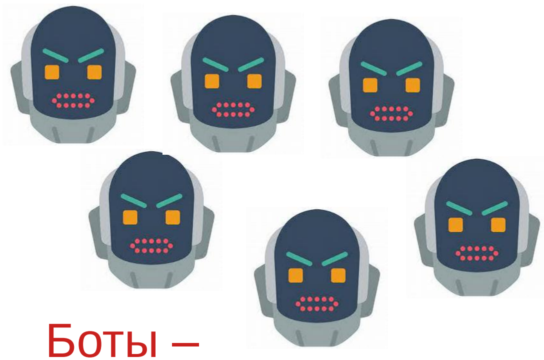
Честные
покупатели

Плохая рейтинговая (репутационная) система

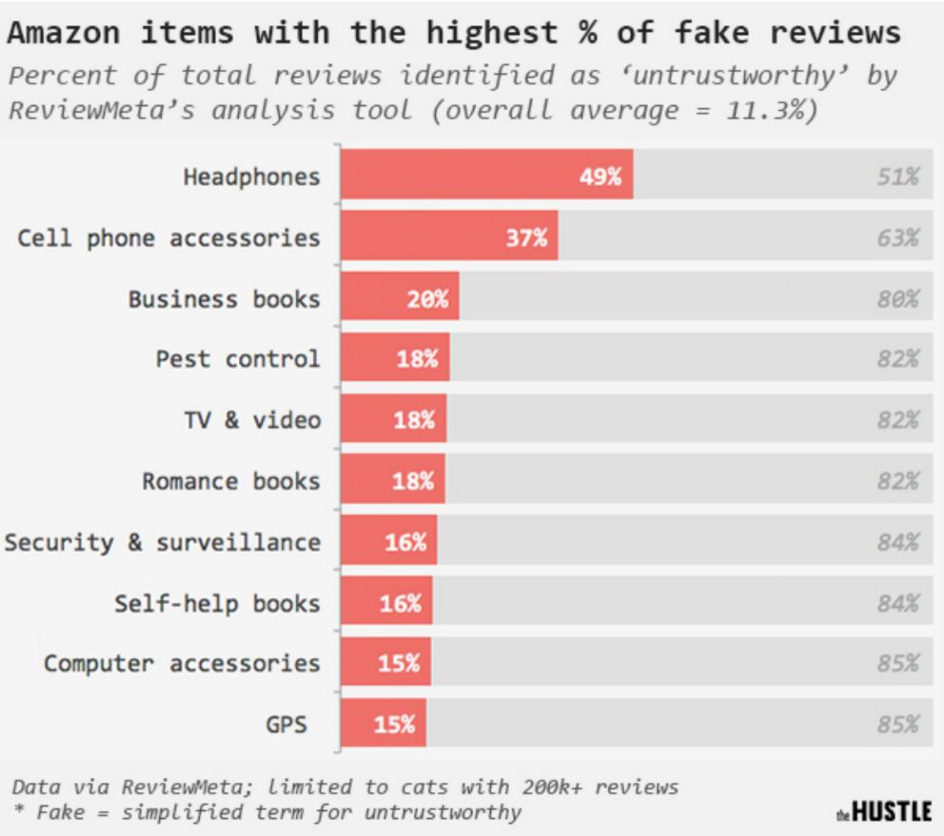


Все маркетплэйсы страдают от “накруток”

Продавец - мошенник



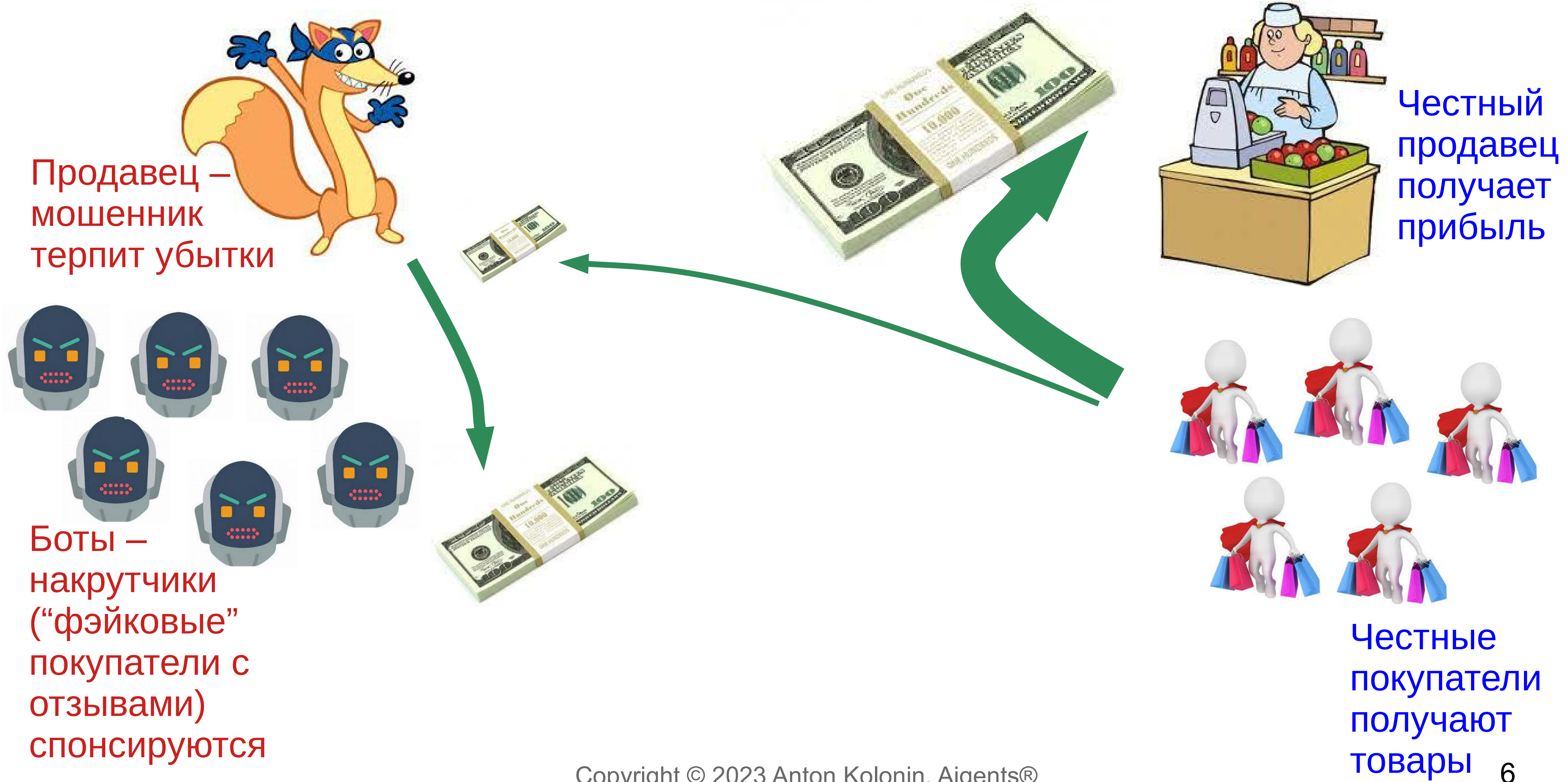
Боты – накрутчики (“фэйковые” покупатели с отзывами)



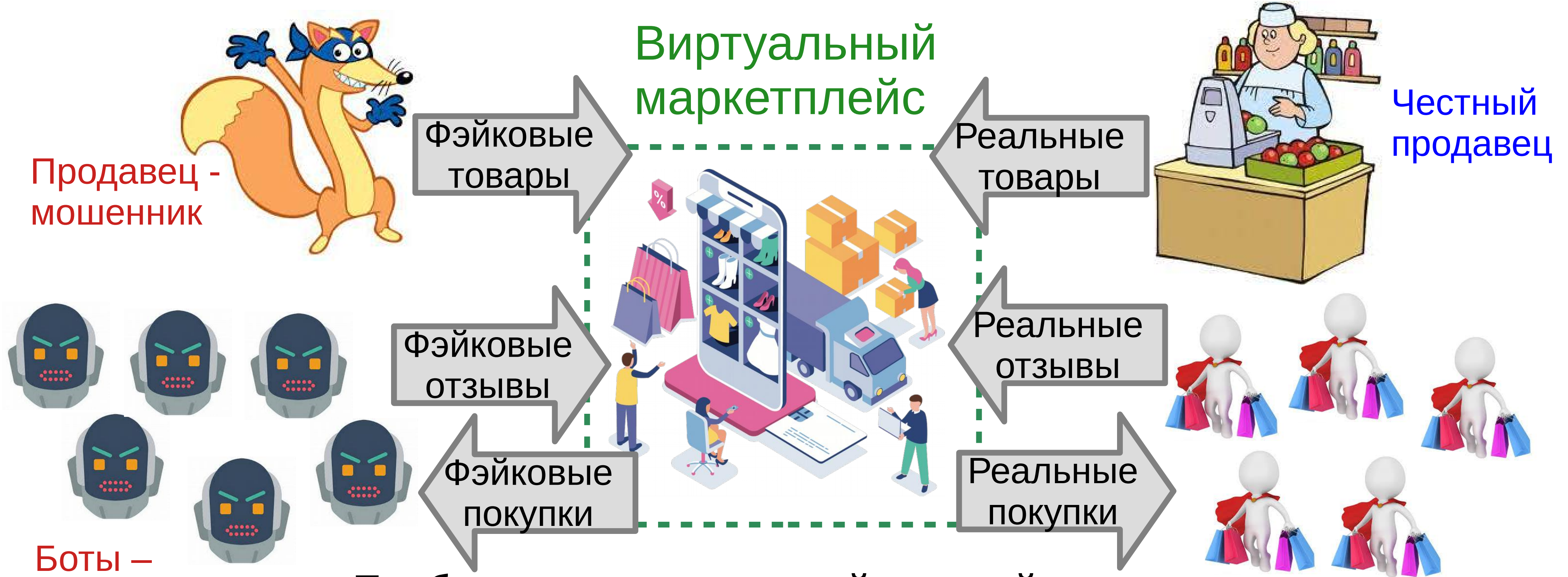
Честный продавец

Честные покупатели

Хорошая рейтинговая (репутационная) система



Имитационное моделирование маркетплейса



Подбираем параметры рейтинговой системы на основе “взвешенного текучего рейтинга”, чтобы максимизировать потери мошенников и минимизировать потери честных покупателей

Репутационная система “взвешенного текучего рейтинга”

Algorithm 1 Weighted Liquid Rank (simplified version)

Inputs:
1) Volume of rated transactions each with financial value of the purchased product or service and rating value evaluating quality of the product/service, covering specified period of time;
2) Reputation ranks for every participant at the end of the previous time period.
Parameters: List of parmeters, affecting computations - default value, logarithmic ratings, conservatism, decayed value, etc.
Outputs: Reputation ranks for every participant at the end of the previous time period.

1: **foreach of** *transactions* **do**
2: **let** *rater_value* be rank of the rater at the end of previous period of default value
3: **let** *rating_value* be rating supplied by trasaction rater (consumer) to ratee (supplier)
4: **let** *rating_weight* be financial value of the transaction of its logarithm, if logarithmic ratings parameter is set to true
5: **sum** *rater_value*rating_value*rating_weight* for every ratee
6: **end foreach**
7: **do** normalization of the sum of the muliplications per ratee to range *0.0-1.0*, get *differential_ranks*
8: **do** blending of the old_ranks known at the end of previous peiod with differential_ranks based on parameter of conservatism, so that *new_ranks = (old_ranks*conservatism+N*(1-differential_ranks))*, using decayed value if no rating are given to ratee during the period
9: **do** normalization of *new_ranks* to range *0.0-1.0*
10: **return** *new_ranks*

- R_d - default initial reputation rank;
- R_c - decayed reputation in range to be approached by inactive agents eventually;
- *C* - conservatism as a blending “alpha” factor between the previous reputation rank recorded at the beginning of the observed period and the differential one obtained during the observation period;
- *FullNorm* – when this boolean option is set to *True* the reputation system performs a full-scale normalization of incremental ratings;
- *LogRatings* - when this boolean option is set to *True* the reputation system applies *log10(1+value)* to financial values used for weighting explicit ratings;
- *Aggregation* - when this boolean option is set to *True* the reputation system aggregates all explicit ratings between each unique combination of two agents with computes a weighted average of ratings across the observation period;
- *Downrating* - when this boolean option is set to *True* the reputation system translates original explicit rating values in range *0.0-0.25* to negative values in range *-1.0* to *0.0* and original values in range *0.25-1.0* to the interval *0.0-1.0*.
- *UpdatePeriod* – the number of days to update reputation state, considered as observation period for computing incremental reputations.

https://github.com/singnet/reputation
https://github.com/aigents/aigents-java/blob/master/src/main/java/net/webstructor/peer/Reputationer.java

https://arxiv.org/pdf/1905.08036.pdf
https://blog.singularitynet.io/minimizing-recommendation-fraud-7dabbee8fc00
https://aiforgood2019.github.io/papers/IJCAI19-AI4SG_paper_28.pdf

Разделение участников рынка на “жуликов” и “честных”

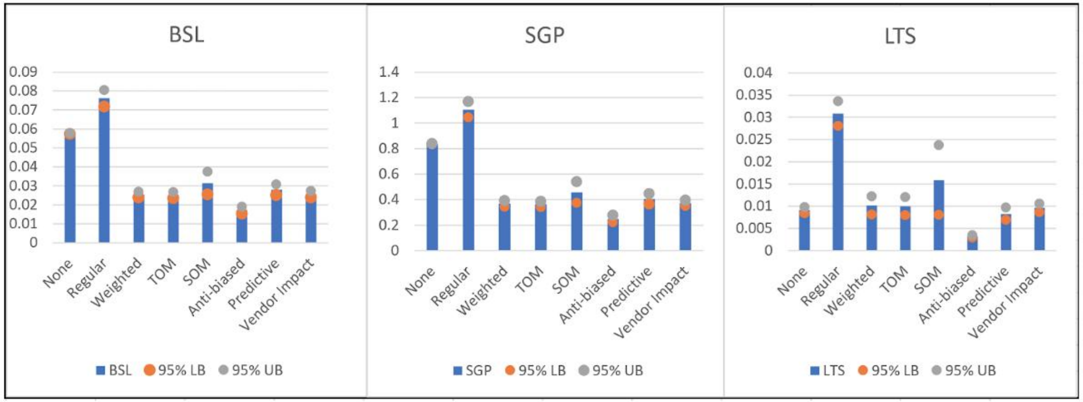
Ranks of Suppliers, dishonest Supplier (including alias) in red and honest suppliers in blue

https://github.com/singnet/reputation
https://github.com/aigents/aigents-java/blob/master/src/main/java/net/webstructor/peer/Reputationer.java

https://arxiv.org/pdf/1905.08036.pdf
https://blog.singularitynet.io/minimizing-recommendation-fraud-7dabbee8fc00
https://aiforgood2019.github.io/papers/IJCAI19-AI4SG_paper_28.pdf

Оптимизация параметров рейтингования в обществе

Reputation System Type	OMU	LTS	BSL	SGP
None	0.99	0.01	0.06	0.83
Regular	0.97	0.03	0.08	1.11
Weighted	0.99	0.01	0.03	0.37
TOM	0.99	0.01	0.03	0.36
SOM	0.99	0.02	0.03	0.46
Anti-biased	1.00	0.00	0.02	0.25
Predictive	0.99	0.01	0.03	0.40
Vendor Impact	0.99	0.01	0.03	0.37



Оптимизируемые социоэкономические метрики

OMU (Organic Market Utility) – органическая полезность рынка

LTS (Loss to Scam) – потери в пользу мошенников

BSL (Buyers Satisfaction Loss) – потеря покупательского удовлетворения

SGP (Seller Gaming Profit) – прибыль от манипуляций

https://github.com/singnet/reputation
https://github.com/aigents/aigents-java/blob/master/src/main/java/net/webstructor/peer/Reputationer.java

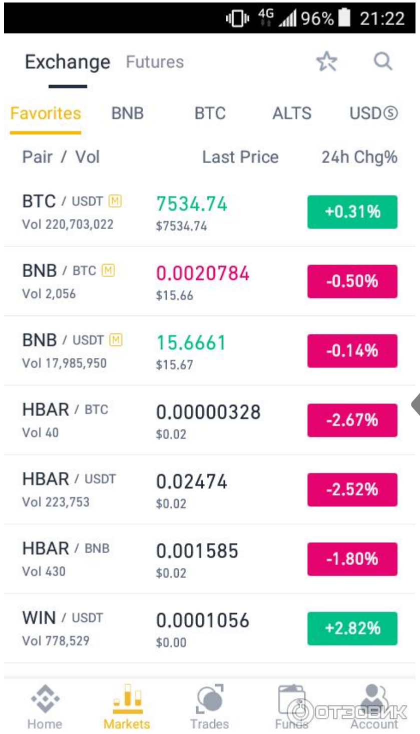
https://arxiv.org/pdf/1905.08036.pdf
https://blog.singularitynet.io/minimizing-recommendation-fraud-7dabbee8fc00
https://aiforgood2019.github.io/papers/IJCAI19-AI4SG_paper_28.pdf

Подбор выигрышных стратегий на бирже

Проигрышная стратегия

Проигрышная стратегия

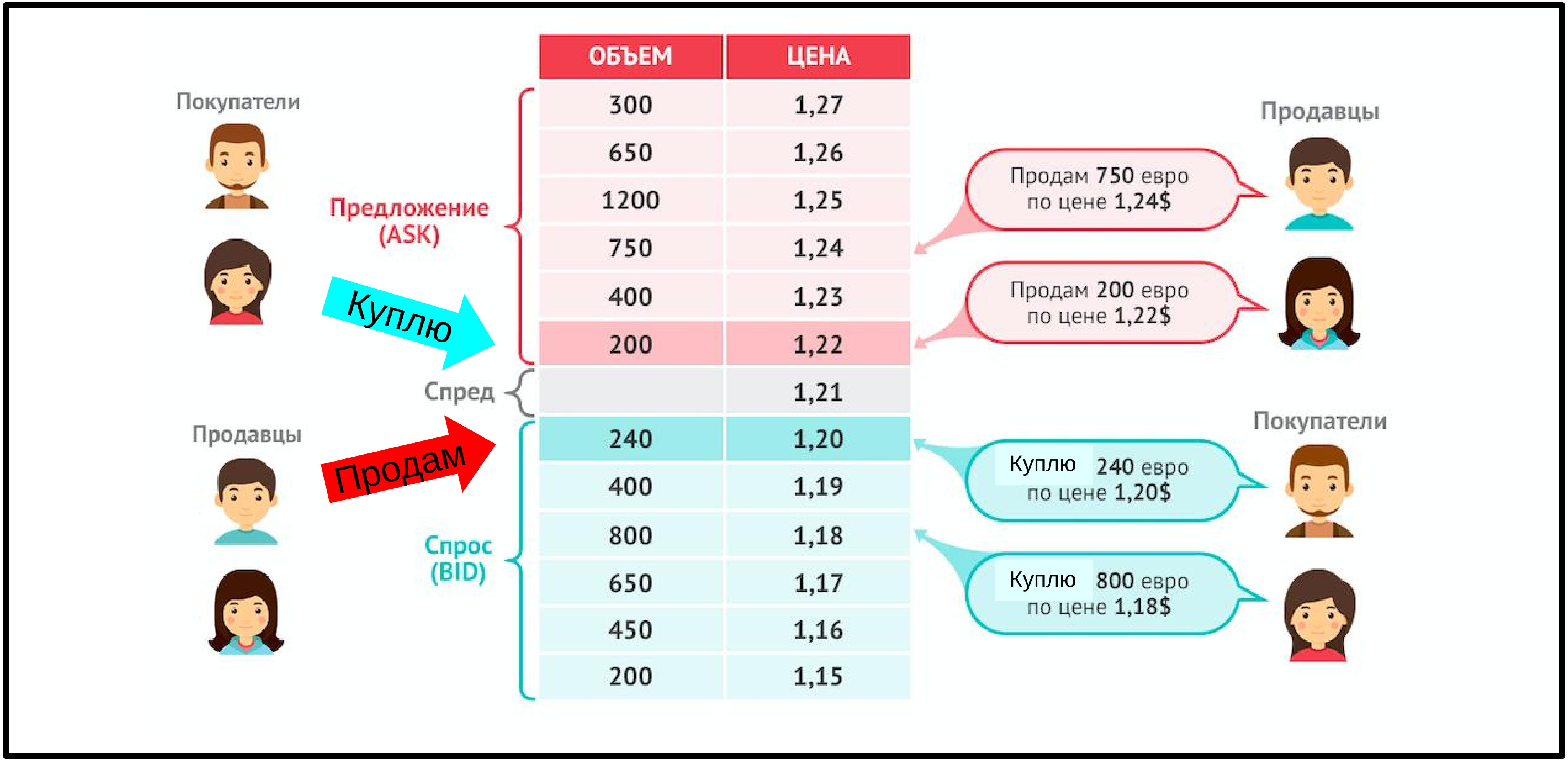
Проигрышная стратегия



Выигрышная стратегия

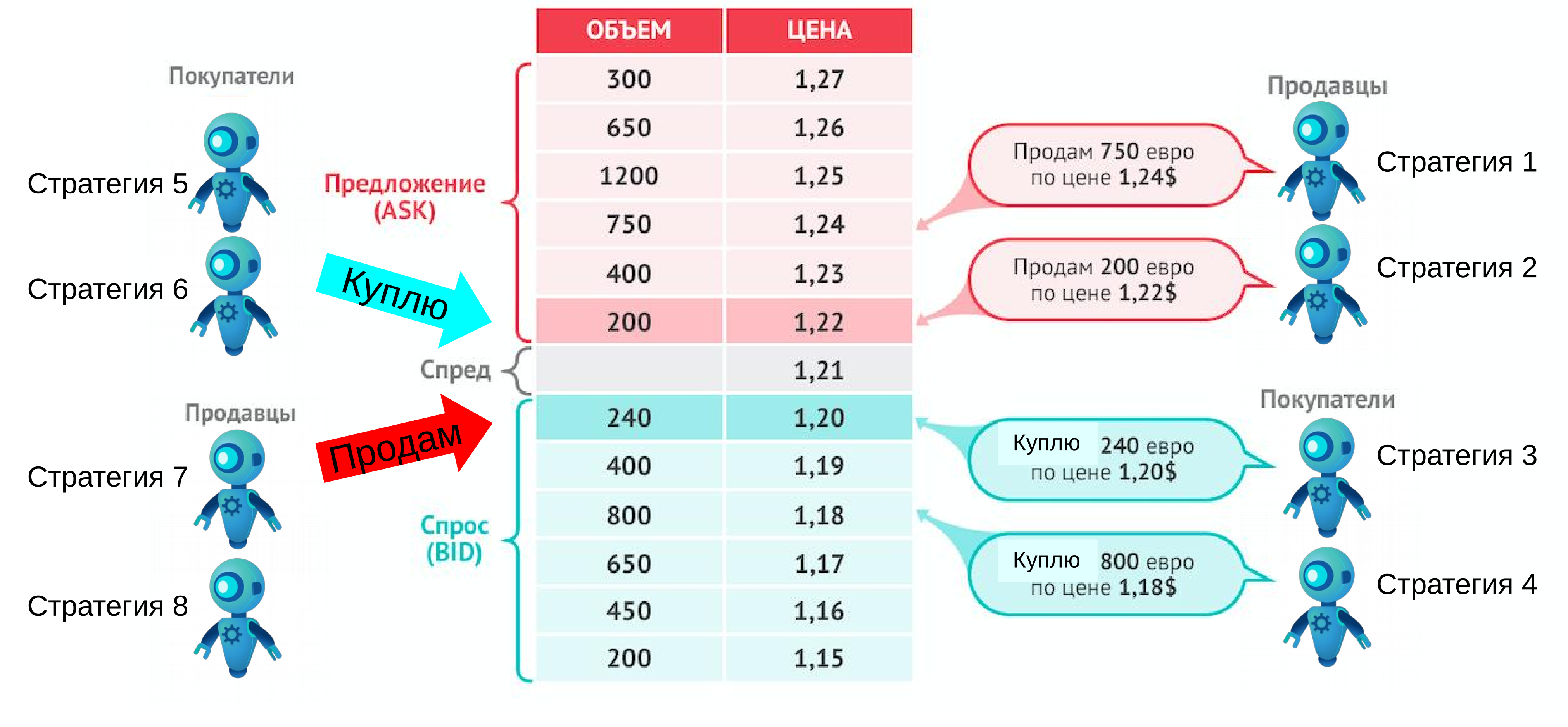
Как подобрать выигрышную стратегию на финансовом (валютном) рынке для себя лично, в ущерб всем остальным (“zero sum game”)?

Реальная работа биржевых торгов

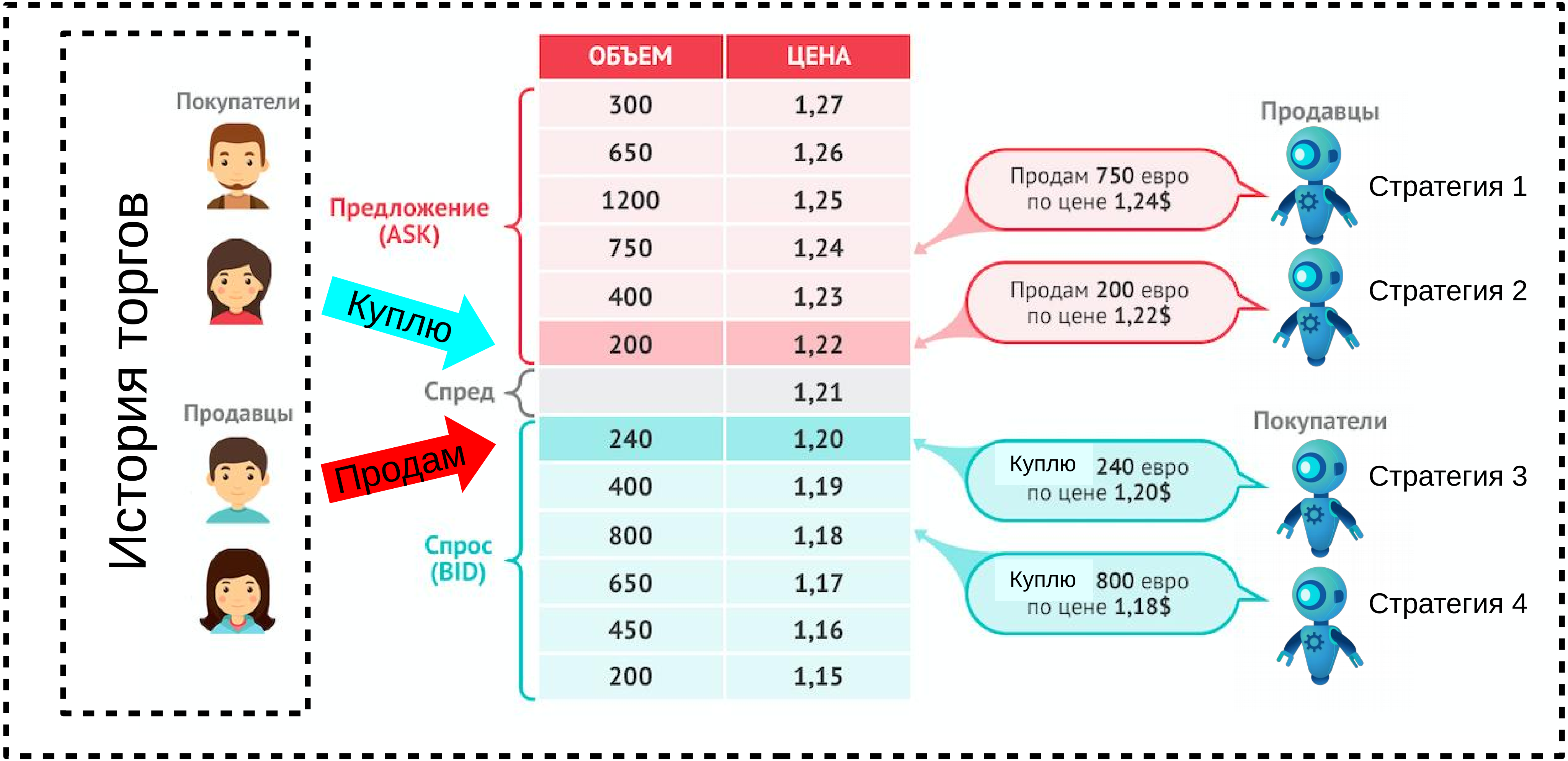


ОБЪЕМ	ЦЕНА
300	1,27
650	1,26
1200	1,25
750	1,24
400	1,23
200	1,22
	1,21
240	1,20
400	1,19
800	1,18
650	1,17
450	1,16
200	1,15

Мульти-агентная симуляция биржевых торгов

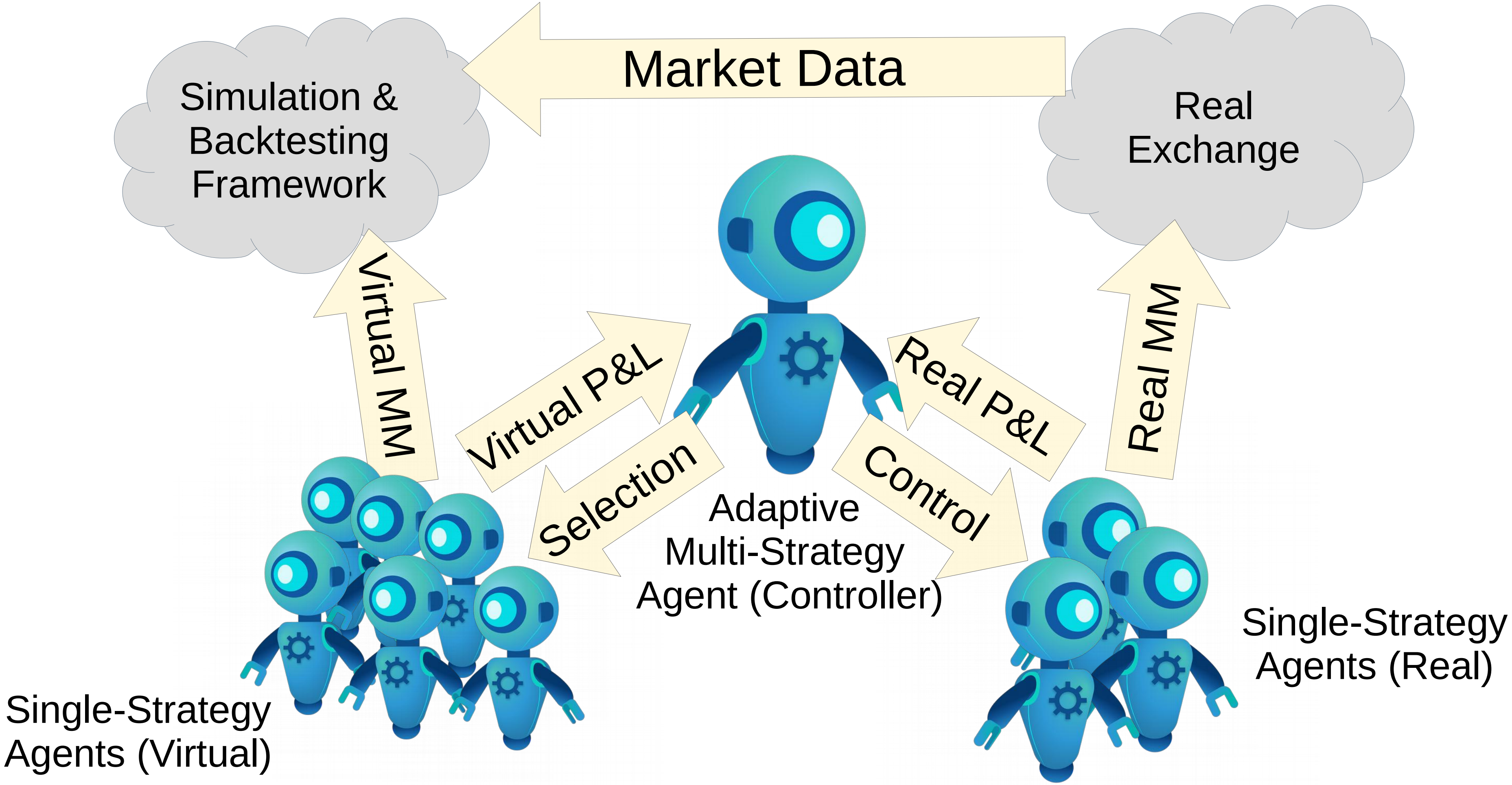


Симуляция торгов на истории (“backtesting”)



ОБЪЕМ	ЦЕНА
300	1,27
650	1,26
1200	1,25
750	1,24
400	1,23
200	1,22
	1,21
240	1,20
400	1,19
800	1,18
650	1,17
450	1,16
200	1,15

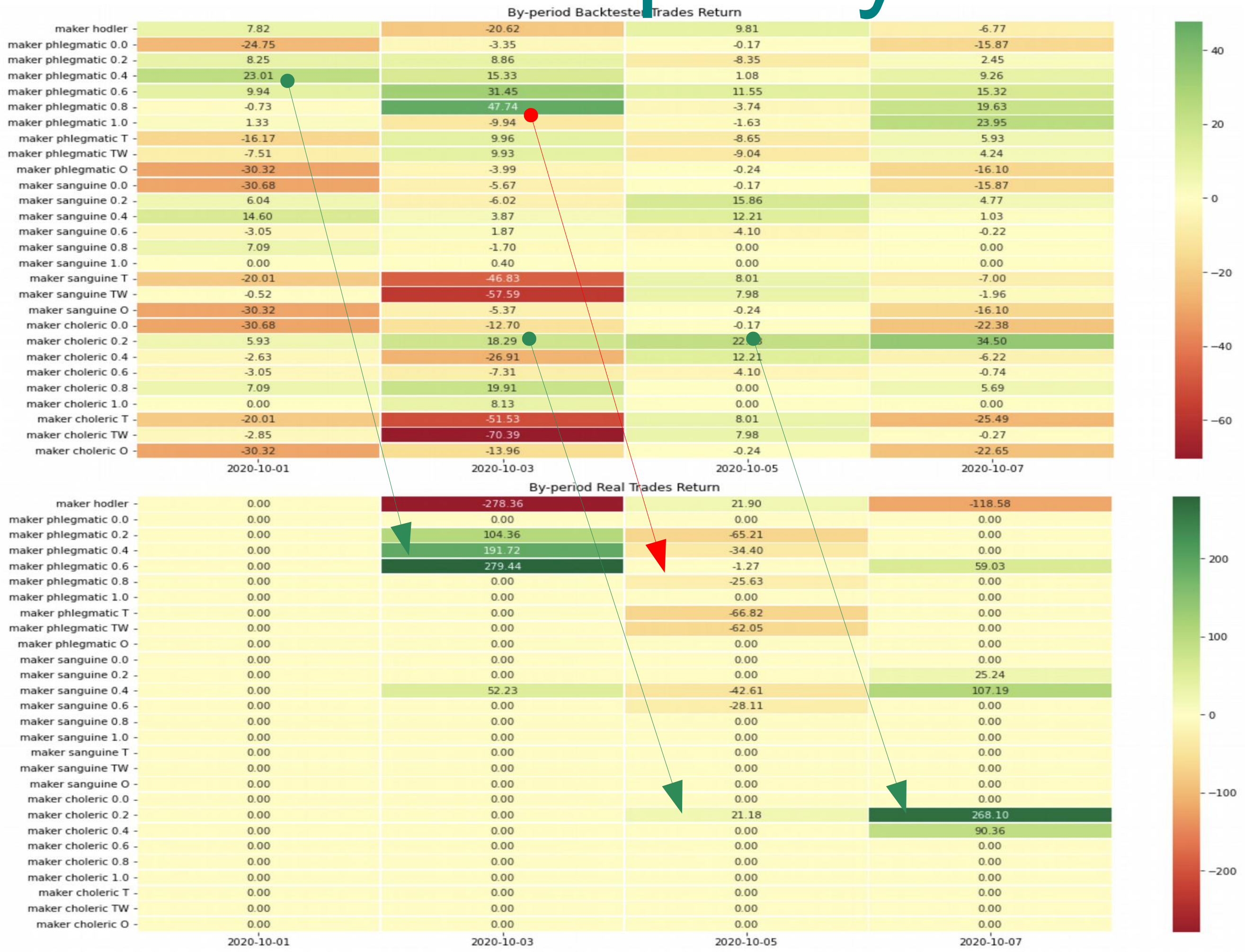
Adaptive Multi-Strategy Agent



Real-time model-based policy selection

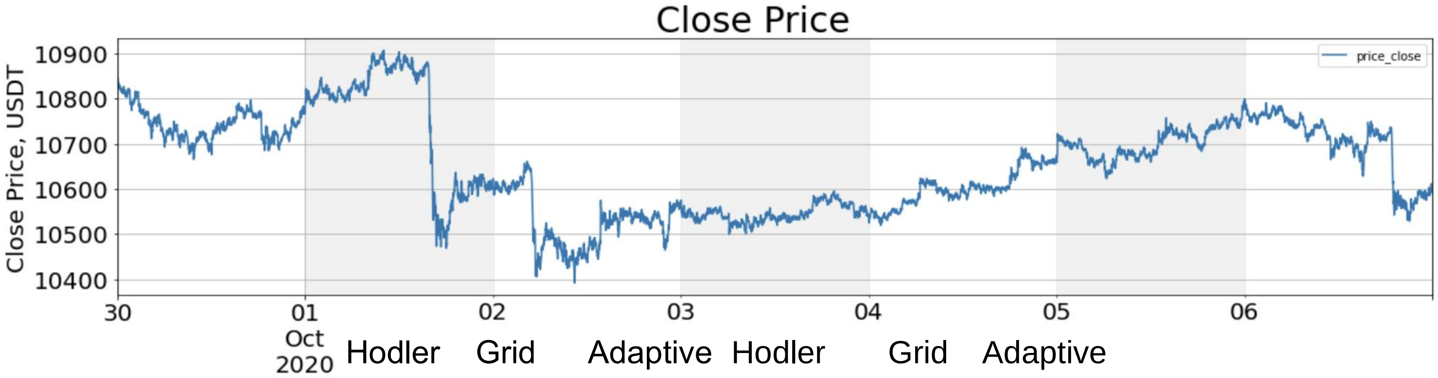
Profit and Loss for Backtesting on historical data / Forward testing on live market data

Profit and Loss for Trading on live market data (still, “backtested”)

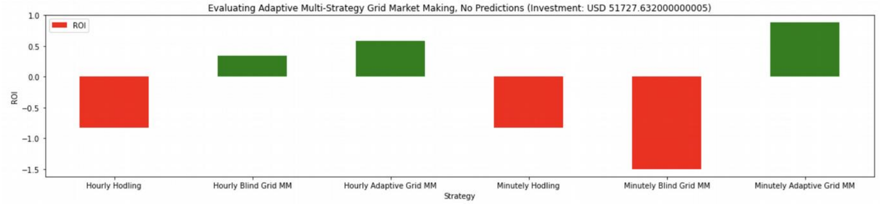


ROI for adaptive MM strategy

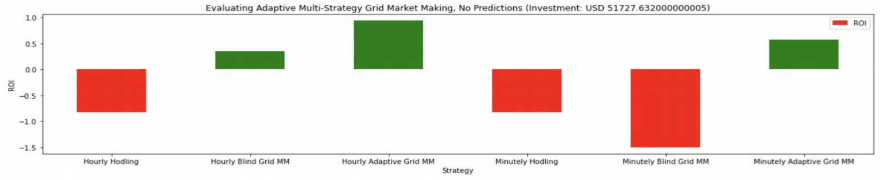
BTC/
USDT



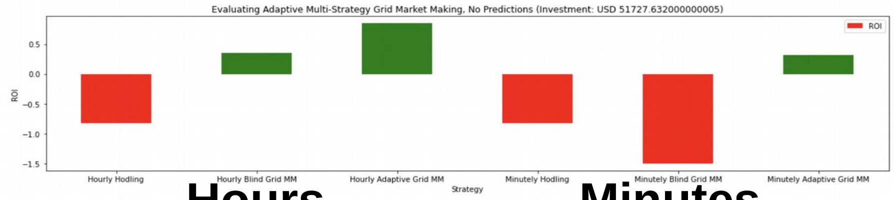
1 day strategy update interval



2 days strategy update interval



3 days strategy update interval



Hours **Minutes**

"Hummingbot"

Pure Market Making

Strategy

1% ROI / 6 days

80% ROI / 1 year

Publications

Architecture of Automated Crypto-Finance Agent
https://ieeexplore.ieee.org/document/9686345
https://arxiv.org/abs/2107.07769

Adaptive Multi-strategy Market Making Agent
https://link.springer.com/chapter/10.1007/978-3-030-93758-4_21

Adaptive Multi-Strategy Market-Making Agent For Volatile Markets
https://arxiv.org/abs/2204.13265

Causal Analysis of Generic Time Series Data Applied for Market Prediction
https://arxiv.org/abs/2204.12928

Social Media Sentiment Analysis for Cryptocurrency Market Prediction
https://arxiv.org/abs/2204.10185

SingularityDAO SingularityNET

Спасибо за внимание!

Антон Колонин

akolonin@aigents.com

Facebook: akolonin

Telegram: akolonin

SingularityNET

AIGENTS

Novosibirsk State University

*THE REAL SCIENCE

https://aigents.com