Experiment 2: KTH treatment

**Model Setup:**

**numPlayers:** 2 (buyer and seller)

**value:** [200, 250, 320]

**cost:** [130, 80, 10]

**numiActions** (Number of individual actions): 9

For both buyer and seller, we label the action (reporting both values and costs) as:

a = 0 report (200, 130) [0, 0]

a = 1 report (200, 80) [0, 1]

a = 2 report (200, 10) [0, 2]

a = 3 report (250, 130) [1, 0]

a = 4 report (250, 80) [1, 1]

a = 5 report (250, 10) [1, 2]

a = 6 report (320, 130) [2, 0]

a = 7 report (320, 80) [2, 1]

a = 8 report (320, 10) [2, 2]

**memory:** 0

**numActions** (number of total actions): 9 \* 9 = 81

**numStates** (number of total states): 9 ^ (2\*0) = 1

**indexActions:** generate an array that can convert the total 81 actions to [x, y] where x is the choice of the buyer (0-8), and y is the choice of the seller (0-8)

For example, the 12-th action will be [1, 3] (the buyer report (200, 80); the seller report (250, 130))

**init\_Prices:** calculate the corresponding prices by the seller’s reporting value and buyer’s reporting cost

|  |  |  |  |
| --- | --- | --- | --- |
| cb\vs | vs=200 | vs=250 | vs=320 |
| cb=130 | 165 | 215 | 285 |
| cb=80 | 115 | 165 | 235 |
| cb=10 | 45 | 95 | 165 |

**init\_Profits:** calculate the profits of the buyers report and the sellers report

return\_size: (numActions, numPlayers) = (81, 2)

the profit for buyers:

the profit for sellers:

for example, if i = 65, by using indexActions, we can get the action of buyer is 65/9 = 7 (320,80) and the action of seller is 65%9 = 2 (200, 10)

vb, cb = 320, 80; vs, cs = 200, 10

the profit for buyers: 200-115-max(0,10-80)=85

the profit for sellers: 115-80-max(0,200-320)=35

**init\_Q:** calculate the Q table

return\_size: (numActions, numiActions, numPlayers) = (81, 9, 2)

As for the initial matrix , the baseline choice is to set the Q-values at t=0 at the discounted payoff that would accrue to player i if opponents randomized uniformly.

For example, assume = 0.95, when a = 0, the choice of buyer is (200, 130)

We can find that no matter the choice of b, the profits of buyer will be 35

Q = (35\*9)/((1-0.095)\*9)=700

At the starting point, all actions have a same matrix as this. Using Q-learning, we will modify this table and find out the optimal state.

Text

Description automatically generated

Experiment 2: SR mechanism

**Model Setup:**

**numPlayers:** 2 (buyer and seller)

**value:** [200, 250, 320]

**cost:** [130, 80, 10]

**numiActions** (Number of individual actions): 27

For the buyer, we label the action by combining the buyer’s actions (reporting both the value and the cost in the initial phase and the value in the arbitration phase) in two phases:

a = 0 first report (200,130), if it goes to arbitration reports 200 [0, 0, 0]

a = 1 first report (200,130), if it goes to arbitration reports 250 [0, 0, 1]

a = 2 first report (200,130), if it goes to arbitration reports 320 [0, 0, 2]

a = 3 first report (200, 80), if it goes to arbitration reports 200 [0, 1, 0]

a = 4 first report (200, 80), if it goes to arbitration reports 250 [0, 1, 1]

a = 5 first report (200, 80), if it goes to arbitration reports 320 [0, 1, 2]

…

a = 25 first report (320,10), if it goes to arbitration reports 250 [2, 2 ,1]

a = 26 first report (320,10), if it goes to arbitration reports 320 [2, 2, 2]

For the seller, we label the action (cost reporting) by

b = 0 first report (200,130), if it goes to arbitration reports 130 [0, 0, 0]

b = 1 first report (200,130), if it goes to arbitration reports 80 [0, 0, 1]

b = 2 first report (200,130), if it goes to arbitration reports 10 [0, 0, 2]

b = 3 first report (200, 80), if it goes to arbitration reports 130 [0, 1, 0]

b = 4 first report (200, 80), if it goes to arbitration reports 80 [0, 1, 1]

b = 5 first report (200, 80), if it goes to arbitration reports 10 [0, 1, 2]

…

b = 25 first report (320,10), if it goes to arbitration reports 80 [2, 2, 1]

b = 26 first report (320,10), if it goes to arbitration reports 10 [2, 2, 2]

**memory:** 0 (only contains the current state)

**numActions** (number of total actions): 27 \* 27 = 729

**numStates** (number of total states): 27 ^ (2\*0) = 1

**buyerInvestment:** 75 (We can fix the investment level from [0, 25, 75])

**sellerInvestment:** 25 (We can fix the investment level from [0, 25, 75])

**trueValue, trueCost:** derived from functions init\_TrueValue, init\_TrueCost, which is based on Table 3

Graphical user interface, text

Description automatically generated with medium confidence

e.g. In this case, the true cost is 10, and the true value is 250

**indexActions:** generate an array that can convert the total 729 actions to [x, y] where x is the choice of the buyer (0-26), and y is the choice of the seller (0-26)

For example, the 120-th action will be [4, 12] (the buyer report (200, 80, 250); the seller report (250, 80, 200))

**init\_Prices:** calculate the corresponding prices by the seller’s reporting value and buyer’s reporting cost

|  |  |  |  |
| --- | --- | --- | --- |
| cb\vs | vs=200 | vs=250 | vs=320 |
| cb=130 | 165 | 215 | 285 |
| cb=80 | 115 | 165 | 235 |
| cb=10 | 45 | 95 | 165 |

**init\_Profits:** calculate the profits of the buyers report and the sellers report

return\_size: (numActions, numPlayers) = (729, 2)

vb, cb, vb\_a are the reported value in the report stage, the reported cost in the report stage, and the reported for buyers respectively

1) if all reports concides,

the profit for buyers:

the profit for sellers:

2) if only the value report differs, the buyer enters into the arbitrage and is fined 300

Text

Description automatically generated

If buyer’s second report is 200, there is no trade

the profit for buyers: 0 - buyer’s investment

the profit for sellers: 0 - seller’s investment

If buyer’s second report is 250,

the profit for buyers: 0.5\*0+0.5\*(true value-205)-buyer’s investment

the profit for sellers: 0.5\*0+0.5\*(205-true cost)-seller’s investment

If buyer’s second report is 320,

the profit for buyers: 0.5\*(true value-205)+0.5\*(true value-255)-buyer’s investment

the profit for sellers: 0.5\*(255-true cost)+0.5\*(205-true cost)-seller’s investment

Notes:

1. The buyer is fined 300
2. if the second report of the buyer matches the first-stage report of the seller, the seller is rewarded 300 else fined 300

3) if only the cost report differs, the seller enters into the arbitrage and is fined 300

Text

Description automatically generated with low confidence

If seller’s second report is 130, there is no trade

the profit for buyers: 0-buyer’s investment

the profit for sellers: 0-seller’s investment

If seller’s second report is 80,

the profit for buyers: 0.5\*0+0.5\*(true value-125)-buyer’s investment

the profit for sellers: 0.5\*0+0.5\*(125-true cost)-seller’s investment

If seller’s second report is 10,

the profit for buyers: 0.5\*(true value-125)+0.5\*(true value-75)-buyer’s investment

the profit for sellers: 0.5\*(125-true cost)+0.5\*(75-true cost)-seller’s investment

Notes:

1. The seller is fined 300
2. If the second report of the seller matches the first-stage report of the buyer, the buyer is rewarded 300 else fined 300

4) if both reports differ, each party has a 50% chance of entering arbitrage and both parties are fined 300

We combine case 2) and case 3) (the bonus and fine when verifying the reported prices in the report stage and arbitrage stage are same)

**init\_Q:** calculate the Q table

return\_size: (numiActions, numPlayers) = (27, 2)

As for the initial matrix , the baseline choice is to set the Q-values at t=0 at the discounted payoff that would accrue to player i if opponents randomized uniformly.

Experiment 3: AM mechanism

In AM mechanism, there is a probability epsilon that arbitration will be called (buyer or seller goes to arbitration with equal probability), regardless of whether there is a match in reported value/cost or not. With probability 1-epsilon, there is no arbitration and the trade goes through (with the price determined as in the SR mechanism) if both the reported value and reported cost match; otherwise there is no trade. The payoff in the arbitration phase is the same as in the SR mechanism.

**Model Setup:**

**numPlayers:** 2 (buyer and seller)

**value:** [200, 250, 320]

**cost:** [130, 80, 10]

**numiActions** (Number of individual actions): 27

For the buyer, we label the action by combining the buyer’s actions (reporting both the value and the cost in the initial phase and the value in the arbitration phase) in two phases:

a = 0 first report (200,130), if it goes to arbitration reports 200 [0, 0, 0]

a = 1 first report (200,130), if it goes to arbitration reports 250 [0, 0, 1]

a = 2 first report (200,130), if it goes to arbitration reports 320 [0, 0, 2]

a = 3 first report (200, 80), if it goes to arbitration reports 200 [0, 1, 0]

…

a = 25 first report (320,10), if it goes to arbitration reports 250 [2, 2 ,1]

a = 26 first report (320,10), if it goes to arbitration reports 320 [2, 2, 2]

For the seller, we label the action (cost reporting) by

b = 0 first report (200,130), if it goes to arbitration reports 130 [0, 0, 0]

b = 1 first report (200,130), if it goes to arbitration reports 80 [0, 0, 1]

b = 2 first report (200,130), if it goes to arbitration reports 10 [0, 0, 2]

b = 3 first report (200, 80), if it goes to arbitration reports 130 [0, 1, 0]

…

b = 25 first report (320,10), if it goes to arbitration reports 80 [2, 2, 1]

b = 26 first report (320,10), if it goes to arbitration reports 10 [2, 2, 2]

**eps:** 0.05 (For simplicity, we fix eps as constant first)

**memory:** 0 (only contains the current state)

**numActions** (number of total actions): 27 \* 27 = 729

**numStates** (number of total states): 27 ^ (2\*0) = 1

**buyerInvestment:** 75 (We can fix the investment level from [0, 25, 75])

**sellerInvestment:** 25 (We can fix the investment level from [0, 25, 75])

**trueValue, trueCost:** derived from functions init\_TrueValue, init\_TrueCost, which is based on Table 3

Graphical user interface, text

Description automatically generated with medium confidence

e.g. In this case, the true cost is 10, and the true value is 250

**indexActions:** generate an array that can convert the total 729 actions to [x, y] where x is the choice of the buyer (0-26), and y is the choice of the seller (0-26)

For example, the 120-th action will be [4, 12] (the buyer report (200, 80, 250); the seller report (250, 80, 200))

**init\_Prices:** calculate the corresponding prices by the seller’s reporting value and buyer’s reporting cost

|  |  |  |  |
| --- | --- | --- | --- |
| cb\vs | vs=200 | vs=250 | vs=320 |
| cb=130 | 165 | 215 | 285 |
| cb=80 | 115 | 165 | 235 |
| cb=10 | 45 | 95 | 165 |

**init\_Profits:** calculate the profits of the buyers report and the sellers report

return\_size: (2, numActions, numPlayers) = (2, 729, 2)

vb, cb, vb\_a are the reported value in the report stage, the reported cost in the report stage, and the reported for buyers respectively

We record the values which do not enter into the arbitrage stage in Profits[0,:,:]

1) if all reports concides,

the profit for buyers:

the profit for sellers:

2) No trade

the profit for buyers:

the profit for sellers:

We record the values which enter into the arbitrage stage in Profits[1,:,:]

1) the buyer enters into the arbitrage

Text

Description automatically generated

If buyer’s second report is 200, there is no trade

the profit for buyers:

the profit for sellers:

If buyer’s second report is 250,

the profit for buyers: 0.5\*0+0.5\*(true value-205)-buyer’s investment

the profit for sellers: 0.5\*0+0.5\*(205-true cost)-seller’s investment

If buyer’s second report is 320,

the profit for buyers: 0.5\*(true value-205)+0.5\*(true value-255)-buyer’s investment

the profit for sellers: 0.5\*(255-true cost)+0.5\*(205-true cost)-seller’s investment

if the second report of the buyer matches the first-stage report of the seller, the seller is rewarded 300 else fined 300

2) the seller enters into the arbitrage and is fined 300

Text

Description automatically generated with low confidence

If seller’s second report is 130, there is no trade

the profit for buyers: 0-buyer’s investment

the profit for sellers: 0-seller’s investment

If seller’s second report is 80,

the profit for buyers: 0.5\*0+0.5\*(true value-125)-buyer’s investment

the profit for sellers: 0.5\*0+0.5\*(125-true cost)-seller’s investment

If seller’s second report is 10,

the profit for buyers: 0.5\*(true value-125)+0.5\*(true value-75)-buyer’s investment

the profit for sellers: 0.5\*(125-true cost)+0.5\*(75-true cost)-seller’s investment

If the second report of the seller matches the first-stage report of the buyer, the buyer is rewarded 300 else fined 300

* We average the profits in 1) and 2) as buyer or seller goes to arbitration with equal probability

**init\_Q:** calculate the Q table

return\_size: (numiActions, numPlayers) = (27, 2)

As for the initial matrix , the baseline choice is to set the Q-values at t=0 at the discounted payoff that would accrue to player i if opponents randomized uniformly.

Experiment 4: SPI mechanism

In SPI mechanism, there is a probability epsilon that arbitration will be called (buyer or seller goes to arbitration with equal probability), regardless of whether there is a match in reported value/cost or not. With probability 1-epsilon, there is no arbitration and the trade goes through (with the price determined as in the SR mechanism) .

**Model Setup:**

**numPlayers:** 2 (buyer and seller)

**value:** [200, 250, 320]

**cost:** [130, 80, 10]

**numiActions** (Number of individual actions): 18

For the buyer, we label the action by combining the buyer’s actions (reporting both the value and the cost in the initial phase and whether accept the offer price in the arbitration phase) in two phases:

a = 0 first report (200,130), if it goes to arbitration refuse offer [0, 0, 0]

a = 1 first report (200,130), if it goes to arbitration accept offer [0, 0, 1]

a = 2 first report (200, 80), if it goes to arbitration refuse offer [0, 1, 0]

a = 3 first report (200, 80), if it goes to arbitration accept offer [0, 1, 1]

…

a = 16 first report (320,10), if it goes to arbitration refuse offer [2, 2 , 0]

a = 17 first report (320,10), if it goes to arbitration accept offer [2, 2, 1]

For the seller, we label the action (cost reporting) by

b = 0 first report (200,130), if it goes to arbitration refuse offer [0, 0, 0]

b = 1 first report (200,130), if it goes to arbitration accept offer [0, 0, 1]

b = 2 first report (200, 80), if it goes to arbitration refuse offer [0, 1, 0]

b = 3 first report (200, 80), if it goes to arbitration accept offer [0, 1, 1]

…

b = 16 first report (320,10), if it goes to arbitration refuse offer [2, 2 , 0]

b = 17 first report (320,10), if it goes to arbitration accept offer [2, 2, 1]

**eps:** 0.05 (For simplicity, we fix eps as constant first)

**memory:** 0 (only contains the current state)

**numActions** (number of total actions): 18 \* 18 = 324

**numStates** (number of total states): 18 ^ (2\*0) = 1

**buyerInvestment:** 75 (We can fix the investment level from [0, 25, 75])

**sellerInvestment:** 25 (We can fix the investment level from [0, 25, 75])

**trueValue, trueCost:** derived from functions init\_TrueValue, init\_TrueCost, which is based on Table 3

Graphical user interface, text

Description automatically generated with medium confidence

e.g. In this case, the true cost is 10, and the true value is 250

**indexActions:** generate an array that can convert the total 324 actions to [x, y] where x is the choice of the buyer (0-17), and y is the choice of the seller (0-17)

For example, the 120-th action will be [6, 12] (the buyer report (250, 130, 0); the seller report (320, 130, 0))

**init\_Prices:** calculate the corresponding prices by the seller’s reporting value and buyer’s reporting cost

|  |  |  |  |
| --- | --- | --- | --- |
| cb\vs | vs=200 | vs=250 | vs=320 |
| cb=130 | 165 | 215 | 285 |
| cb=80 | 115 | 165 | 235 |
| cb=10 | 45 | 95 | 165 |

**init\_Profits:** calculate the profits of the buyers report and the sellers report

return\_size: (2, numActions, numPlayers) = (2, 324, 2)

vb, cb, vb\_a are the reported value in the report stage, the reported cost in the report stage, and the reported for buyers respectively

We record the values which do not enter into the arbitrage stage in Profits[0,:,:]

* if no one into arbitrage,

the profit for buyers:

the profit for sellers:

We record the values which enter into the arbitrage stage in Profits[1,:,:]

* if someone into arbitrage

1) the buyer enters into the arbitrage and is fined 300, offered a price at vb+5

If buyer refuse the offer, there is no trade, seller is fined 300

the profit for buyers:

the profit for sellers:

If buyer accept the offer, trade occurs, seller is awarded 300

the profit for buyers:

the profit for sellers:

2) the seller enters into the arbitrage and is fined 300, offered a price at cs-5

If seller refuse the offer, there is no trade, buyer is fined 300

the profit for buyers:

the profit for sellers:

If seller accept the offer, trade occurs, buyer is awarded 300

the profit for buyers:

the profit for sellers:

* We average the profits in 1) and 2) as buyer or seller goes to arbitration with equal probability

**init\_Q:** calculate the Q table

return\_size: (numiActions, numPlayers) = (18, 2)

As for the initial matrix , the baseline choice is to set the Q-values at t=0 at the discounted payoff that would accrue to player i if opponents randomized uniformly.