

## **Judo Bots**

In the case “Judo in Action”, you saw several examples of companies that managed to make a virtue out of staying small in markets dominated by large incumbents. In this experiment, your job is to devise or defend against a judo strategy of an entrant.

### **Market Structure**

There are, potentially, two firms competing in this market, an incumbent, who has established operations, a known brand, and established cost structure, and, possibly, an entrant with none of these things.

The incumbent’s product is known to all consumers in the market. The incumbent has variable costs of \$100/unit to make the product and no fixed costs. The product is made to order, so inventories and the like are of no consequence.

Currently, only the incumbent serves the market, but an entrant can set up shop by paying an up-front set-up cost equal to \$500. Once it sets up shop, its cost structure is determined. The entrant is attempting a new production technology, whose effectiveness cannot be determined in advance. With 50% technology, its innovation succeeds and its variable costs are only \$80/unit. Otherwise, it fails and it merely mimics the incumbent at \$100/unit variable costs. As an entrant, you will know your costs of production, but the incumbent will not. Denote the realized variable cost by  $c$ .

### **Consumers and Advertising**

Both incumbent and entrant seek to attract a market consisting of 100 consumers, each of whom has unit demand. Consumers value the incumbent’s product at \$200 per unit. The value placed on the entrant’s product cannot be determined in advance. With 50% probability, they view the two products as perfect substitutes and value the entrant’s product at \$200. Otherwise, they perceive the entrant’s product as inferior and value it at \$160. Neither the incumbent nor the entrant will know the value that consumers place on the entrant’s product until demand is realized.

Consumers are initially unaware of the entrant’s product, but it can produce informational advertising to alert  $N$  of the 100 consumers, where  $N$  can be any integer from 0 to 100. It costs the entrant nothing to engage in this advertising. Competitor intelligence on the part of the incumbent allows it to know exactly how many consumers were targeted when making pricing choices.

### **Buying Choices**

When deciding whether and from whom to buy, consumers do the following:

- a) If a consumer only knows about the incumbent’s product, she buys from the incumbent.

- b) If a consumer is aware of both products, she buys from whichever competitor offers higher surplus. Specifically, she buys from the entrant if her willingness to pay for the entrant's product minus the entrant's price > willingness to pay for incumbent's product - incumbent's price. In the event of a tie, she buys from the incumbent.

## Timing

The timing of the game is as follows:

Step 1: Entrant sets up shop or not.

Step 2: If entrant set up shop, it learns its costs and then chooses a price  $p \leq 200$  and a number of consumers to target  $n$ , an integer from 0 to 100 inclusive.

Step 3: Whether the entrant entered and, if so, the values of  $p$  and  $n$  are revealed to the incumbent. At this point, the incumbent chooses its price,  $P \leq 200$ .

Step 4: Based on buying choices, consumers make purchase decisions. Let  $k$  be the number of consumers buying from the entrant and  $100 - k$  the number buying from the incumbent.

## Profits

$$\text{Incumbent} = (P - 100) \times (100 - k)$$

$$\text{Entrant} = k \times (p - c) - 500$$

$$\text{Entrant} = 0$$

if entered

if not entered

## Playing the Game

To play this game, you will design simple robots choosing strategies for the roles both of incumbent and entrant. Each robot will play every other robot *once* in each role (i.e. team A's bot plays B's once as the incumbent and once as the entrant). The computer will randomly generate the variable costs and willingness to pay associated with the entrant for each run of the game. Your ecu payoffs are equal to the total amounts obtained over all matchups.

### *Survivor Scoring*

We will also run a special version of the simulation purely for Survivor purposes. In this version of the Sim, only the remaining Survivor teams will compete. In addition, for this version of the simulation, the computer will randomly generate the variable cost and willingness to pay associated with the entrant *exactly once*. These values will be used for *all* runs of the simulation. Thus, if it so happens that the computer produces low marginal cost and high WTP for the rival, this will be true whenever any bot plays the rival role. In other words, lucky draws for costs and WTP play no role in scoring the game for Survivor. Each of the seven teams plays every other team once, in both roles.

The highest scoring team gets immunity. All others go to tribal council. If there is a TIE for high score then *no one gets immunity*.

### Designing Bots

By Weds at midnight, each team must submit one entrant bot and one incumbent bot.

To design an entrant bot, please complete the following form:

1. Entry decision: Yes/No
2. If variable costs = 80, then
  - a. Price =  $p_{80}$
  - b. # of consumers targeted =  $n_{80}$
3. If variable costs = 100, then
  - a. Price =  $p_{100}$
  - b. # of consumers targeted =  $n_{100}$

All prices must be  $\leq 200$ ; consumers targeted =  $\{0, 1, 2, \dots, 100\}$ .

To design an incumbent bot, the following format is required:

- Restrict yourself to the following pricing strategies (all others are dominated):
  - a.  $P = 200$
  - b.  $P = p + 40$  (i.e. beat rival when you have WTP advantage, lose targeted customers if not)
  - c.  $P = p$  (i.e. beat rival always)

Note that you will not know whether you have WTP advantage, so prices cannot depend on this.

### Please fill triggers for each strategy.

Example Bot 1:

$P = 200$  if  $p < 130$

Else  $P = p$

Example Bot 2:

$P = 200$  if  $p \times n < 800$

$P = p + 40$  if  $p \times n > 800$  and  $p \times n < 1000$

$P = p$  otherwise

Example Bot 3:

$P = 200$  if  $p < 150$  or  $n < 30$

$P = p + 40$  if  $p > 150$

$P = p$  otherwise

Or you can do weird things like:

Example Bot 5:

$P = 200$  if  $n$  is Fibonacci and ( $p > 127$  and  $p < 151$ )

$P = p + 40$  if  $n$  is Fibonacci and ( $p < 127$  or  $p > 151$ )  
 $P = p$  otherwise

The point of the weird example is to show that the contingencies can be sets of prices with gaps in them or non-consecutive sequences of targeted consumers.

Details: If the entrant does not enter, all incumbent bots choose  $P = 200$  automatically. If  $p + 40 > 200$ , all bots automatically charge \$200 since this is the maximum feasible price.

*All bot submissions are due Weds at midnight.* Survivor results will be out by noon on Thursday. Other results will be shown later on Thursday. Tribal Council will be held at its usual time on Friday.