Simulation models for economics

Andrea Nettuno, Riccardo Preziusi, Gianluca Ronga

2015 07 18

Trading dynamics in a controlled market

We are going to create and to develop a NetLogo model in which the basic idea is the representation of a real market: there is a variable number of companies, let's say the incumbents, which try to sell their products characterized by specific features. In order to get the target, some of them will decide to collude in order to increase their profits by applying higher prices and by increasing their surplus.

The anti-competitive behaviour of the firms is prevented by the "third counterpart" between firms and consumers: the Antitrust. In order to get a realistic view of the effects and consequences on the patch, we will generate a rational demand side able to select the best choice. First of all, it is necessary to create some breeds which, in our case, are three: firms, Antitrust and consumers whose features will be better explained in the following part of this note (firms-own and so on). Moreover, there is the possibility to create links among firms, in order to represent their trade and collusion, but there are also yellow links that show the Antitrust's control. Now, it is important to understand the core of the project.

THE DESCRIPTION OF THE MODEL AND OF THE CODE

First of all, as it is said above, the model is characterized by three breeds:

```
breed [firms firm]
breed [antitrusts antitrust]
breed [consumers consumer]
```

All breeds have particular features which will be better explained during this presentation.

```
firms-own [profits minCosts collusionLevel creativity preference matches dossiers price]
links-own [duration]
consumers-own [preference]
```

Then with the button "*setup*", it is possible to obtain the static part of the model, whereas with the button "*go*" the dynamic part.

```
to setup
ca
enterprise
selection
create-antitrust
create_consumers
```

BUSINESS ACTIVITY: FIRM-SIDE

With the procedure "*enterprise*" it is possible to set up firms whose number depend on a slider called "*Nfirms*" and one of their most important characteristic is that they show a variable number ("*CollusionLevel*" that is random from 0 to 4) which is their tendency to collude: if they have 4/4, there is much more probability that they generate a 'secret' link with another company in order to increase their

profits and their color is red. However, if their collusion level is low (e.g. Lower than 3) their color is green.

As it is possible to see below, the agents are set with a factory shape and they are randomly put on the patch. It is interesting to notice that both *minCosts* and profits depend on sliders. The idea behind this illegal behaviour is the possibility to increase returns: the higher is the collusion effect, the higher is the market power shared and the higher will be the surplus gained by firms to the detriment of the clients. The idea is to provide also to identified illegal behavior a penalty proportional to the surplus and to the abuse of market power.

The activity of the firms is also based on trade and their profits depend on the number of links they create with other companies and on an initial level which is regulated by a slider ("*initProf*").

```
to enterprise
  create-firms Nfirms [set shape  "factory"
    setxy random-xcor random-ycor
    set size 1
   set collusionLevel random 5
   set minCosts (AvgminCost + random 10)
  set preference random 5
  set price 1
  set profits initProf
  ]

end
```

After being created, firms select themselves with respect to their CollusionLevel and start trading.

```
to selection
  ask firms [ifelse collusionLevel > 3 [set color red] [set color green]]
end

to trade
ask one-of firms [let counterpart one-of firms if counterpart != self [create-trading-with counterpart]
  set profits profits + count my-in-links with [color = white]]

end
```

The command "*passTime*" is something like a clock accounting for the time that passes by and it implies that links can not exceed a maximum value of duration, as if this duration was the length of some contract. When the link duration exceed the maximum duration these links die. The value of max duration depends on a slider.

```
to passTime
  ask links [set duration duration + 1]
ask tradings [if duration > maxDuration [die]]
ask controls [if duration > maxDuration [die]]
ask firms [set matches 0]
end
```

The collusion activity is explained by the command "*collude*" and it is referred to links among companies based on their collusion level and so, as it is said before, their probability to collude. Companies with CollusionLevel set to zero do not collude.

```
to collude2

  ask firms [ if count firms with [collusionLevel = 1] != 0 [ask one-of firms with [collusionLevel = 1] [if random 100 < 5
        [let myMate one-of firms with [collusionLevel > 2] if myMate != self and myMate != nobody
      [create-collusion-with myMate [set color red] set price price + 2 ]]]]]

  ask firms [ if count firms with [collusionLevel = 2] != 0 [ask one-of firms with [collusionLevel = 2] [if random 100 < 10
        [let myMate one-of firms with [collusionLevel > 2] if myMate != self and myMate !=  nobody
      [create-collusion-with myMate [set color red] set price price + 2 ]]]]]

  ask firms [ if count firms with [collusionLevel = 3] != 0 [ask one-of firms with [collusionLevel = 3] [if random 100 < 30
        [let myMate one-of firms with [collusionLevel > 2] if myMate != self and myMate != nobody
      [create-collusion-with myMate [set color red] set price price + 2] ]]]]

  ask firms [ if count firms with [collusionLevel = 4] != 0 [ask one-of firms with [collusionLevel = 4] [if random 100 < 50
        [let myMate one-of firms with [collusionLevel > 2] if myMate != self and myMate != nobody
      [create-collusion-with myMate [set color red] set price price + 2]]]]]


end
```

Since, there exist companies with collusion level set equal to one then there is a probability of five per cent that this firm colludes with an other one with collusion level higher than two and so on. The idea is that since the collusion level is higher, higher is the probability of colluding and to look for other companies willing to collude.

Moreover, as far as the collusion activity is concerned, it is possible to refer to the command "*cheat*". This command tries to reply the incentive to cheat of colluding firms and it is related to a switcher which enable or disable the incentive to cheat. The idea to test if incentive to cheat has some effect on aggregate profits made by colluding firms and if it may be a cause of their fluctuations. The procedure is structured in this way: an already colluding firm will break their collusive links with 10% probability and will get a surplus that is randomly set between 1 and 50. In the code there is also another cheat procedure, cheat2, which is a trial to understand if the incentive to cheat may be inversely proportional to the surplus illicitely gained, but the experiments showed no evidence of the fact.

Finally, the "*threshold*" procedure is set to avoid negative prices. If price is less than 1 it is set equal to 1. What's more since during their normal activity firms bear costs and exit from the market, if profits of our firms (here used also as a variable describing cash-flows) are lower than their minCosts, firms will be excluded from the market. This to create a little dynamic-competition.

```
to threshold

  ask firms [if price < 1 [set price 1]]
  ask firms [if profits < minCosts [die]]

end
```

ANTITRUST ACTIVITY

The market is controlled by a a third counterpart (*ANTITRUST*) which monitors the companies in the market. The basic idea is that Antitrust randomly controls the enterprises for a fixed number of ticks (*duration*) and that the function of Antitrust itself depends on the number of control (which is, in this case, a slider "*AntitrustCTRLS*") and the control is represented by a yellow link between the two parts. In this model, the Antitrust is represented as a yellow pentagon in the centre of the patch.

```
to create-antitrust
create-antitrusts 1 [set color yellow
    setxy 0 0
    set size 1.5
    set shape "pentagon"]

end
```

Controls are created every tick by using a temporary variable definition, “suspect”, which represents the firm that is under control and is suspected to collude.

```
to control1

   ifelse AntitrustAct
   [ask antitrusts [let suspects n-of AntitrustCTRLS firms create-controls-to suspects [set color yellow]]]
   []


end
```

When a company is caught colluding while it is under control, situation described by a firms having both yellow and red links coming to its position, Antitrust gives a penalty to the colluding firm under control and the penalty may be either lump sum or proportional to the advantages obtained illicitly by firms (e.g. average of profits), then the agency remove the guilty links.

In this model, as it is possible to see below, with the command “fine”, a caught colluding company loses part of its profits and this depends on how much higher is the fine (here expressed by a slider “FinesLevel).

The command “fine2” is characterized by introducing the mean of green companies' profits (firms with low CollusionLevel) and finally with “fine3” it is created a dossier which represents the memory of the Antitrust. The dossier is another way Antitrust have to get an estimation of the profits illicitely gained by colluding firms. Always creating a temporary variable (G) with the “let” command, the fine chosen by the Antitrust to restore competition is based on a dossier created in the first day of control, that is multiplied by the number of day a firms is considered suspect. Of course this procedure raise problems of underestimation and overestimation of the right value of the fine to be attributed to firms, but it is a trial to get to the “optimal fine level”.

```
to fine

  ifelse Lump_sum [
  ask firms [let G one-of firms with [count my-in-links with [color = yellow] != 0] if G != nobody
    [ask G [if count my-in-links with [color = red] != 0 and profits > FinesLevel
       [set profits profits - FinesLevel]     ;; lump sum fines
  ask G [ask my-links with [color = red] [die] ] ]]]]
  []


to fine2

  ifelse Mean_profits [
  ask firms [let G one-of firms with [count my-in-links with [color = yellow] != 0] if G != nobody
    [ask G [if count my-in-links with [color = red] != 0 and profits > FinesLevel
       [set profits profits - (mean [profits] of firms with [color = green])]
  ask G [ask my-links with [color = red] [die] ] ]]]]
  []

end


to fine3

 ifelse Dossier [


 ask firms [let G one-of firms with [count my-in-links with [color = yellow] != 0] if G != nobody
   [ask G [if count my-in-links with [color = red] != 0 and profits > dossiers * MaxDuration
       [set profits profits - (dossiers * MaxDuration + random 20)]
  ask G [ask my-links with [color = red] [die] ] ]]]]
;;  [set profits profits - (dossiers * count my-in-links with [color = red]]

 []

end
```

The Antitrust's activity is also related to the duration which represents the enforcement of the Antitrust itself. It is interesting the presence of a switcher "AntitrustAct" which allows to understand what happens in the model with and without the Antitrust (better explained below on the experiment part of the note). Finally, there are switchers also for choosing what type of fines the Antitrust uses and to enable the incentive to cheat.

ENTRY BARRIERS

It is important to consider the entrance of new companies into the market. New firms have higher entrance cost (avgminCost + 5 + random 10) and are more likely to

exit from the market in few ticks than already existing ones. In order to survive, they have to show either a high level of innovation that, in this model, is called creativity or a good cost structure which will allow the firm to survive in the competitive market. In particular, if that level is higher than a fixed number (e.g. 2, if it is considered a random 3), they can survive in the market i.e. they appear on the patch, with a different colour, the blue, with respect to the initial ones since their creativity is set equal to 2. The other features are the same of the initial firms.

```
to enter
  ask one-of firms [if random 100 < 1 [hatch-firms 5 [setxy random-xcor random-ycor
      set minCosts (avgminCost + 5 + random 10 )
      set profits initProf
      set collusionLevel random 5 ifelse collusionLevel > 3 [set color red][set color green]]


      set creativity random 3 if creativity = 2 [set color blue]]

  ]
  end
```

The incumbent firms could acquire the new profitable start-ups and, in this model, it is explained by the command “purchase”. An existing company can decide to acquire a new entrance in order to obtain a cheaper innovation or technology. In the model, it is possible to see that the company which purchases the new firm is set white whereas the acquired firm (with creativity 2) is blue. It is imposed a link of blue color if one firm is willing to purchase an innovative firm, an acquisition agreement and it can happen only the profits of the acquiring company are higher than profits of the acquired firm and linked to a 10% probability to sign the deal.

```
to purchase
  ask one-of firms [if count firms with [creativity = 2] != 0 [let B one-of firms with [color = blue]
      if random 100 < 10 and B != nobody and [profits] of B < [profits] of self
      [create-agreement-with B [set color blue] set profits profits + random 40 - [profits] of B set creativity 2 set color white
        ask firms with [color = blue] [if count my-in-links with [color = blue] != 0 [die]]]]]]



end
```

THE DEMAND SIDE

In order to introduce the demand side of our market, it is necessary to create another breed: consumers.

```
to create_consumers

  create-consumers 2000
  ask consumers [
    setxy random-xcor random-ycor
   hide-turtle
   set preference random 5
   set size 0 ]


end
```

```
to move

  ask consumers [fd random 5 rt random 360]

end
```

The idea is to put in relation firms and clients. It is introduced the consumer-own variable "preference" and if both company and costumer have the same preference, than firms sell their goods to these clients. In this model there are 2000 invisible consumers on the patch who can move and find companies with their same preference level.

```
to gain-and-pay
  ask firms [let myBuyer one-of consumers in-radius 3 if  count my-in-links with [color = red] != 0
    [if [preference] of self = [preference] of myBuyer
      [set matches matches + 1 set profits (profits + matches * price) - AvgminCost set dossiers (matches * price) ]]
```

It is possible to see that consumers decide to purchase products of a firm not only making reference to their preference levels, but also if this firm is near to them. Indeed, costumers buy from a company if it is inside a given radius.

The boicot function is very useful and explains the case where prices are too high, signal of an hidden collusive activity which drives prices up. In this situation consumers decide not to buy when price is higher than five then firm react by ending their collusion agreement as a consequence of the collapsing demand and firms have

to reduce prices until one, the initial price level.

```
to boicot

  ask consumers [let mySeller one-of firms if mySeller != nobody and [price] of mySeller > 5
    [ask mySeller [ask my-links with [color = red] [die] set price 1]]]


end
```

An other very interesting command is “Set-trasparency” which represents only the instantaneous links and it is taken from a library model by Uri Wilenski, "transparency example”.

COLOUR ANALYSIS OF THE MODEL

In order to simplify the interaction between turtles and the code, to make everything more comprehensible a graphical key is used in order to be able to distinguish at each tick the dynamism of the market.

Red firms are firms with *collusionLevel* higher or equal to 4.

Green firms are firms with collusionLevel equal to 3 or lower.

Blue firms are firms with high potential of innovation, i.e. *Creativity* = 2. White firms are firms that purchased an high-innovation firm.

The Antitrust can be recognized by its yellow colour.

Links colour has a significance as well. What's more, we created different link-breeds to identify them and to give them special features, as for example the fact that only trading links and controls have a fixed duration.

```
directed-link-breed [controls control]
undirected-link-breed [tradings trading]
undirected-link-breed  [collusions collusion]
undirected-link-breed [agreements agreement]
```

White links are “pure”. They only represent the simple trading between firms, that is another source of profits. Red links identify collusive relationships. To be noticed that green firms can create collusive links as well, only the ones with collusionLevel set at zero will not do it. Yellow links identify the controls made by the Antitrust. Blue links describe an acquisition agreement. When the graphics show the blue link it means that an acquisition agreement between a blue firm and another firm is in progress, then it may conclude positively or not.

In the end another very interesting command is “Set-trasparency” which represents only the instantaneous links and it is taken from a library model by Uri Wilenski, "transparency example”. It is a trial to avoid the confusion created by too many links on the display.

EXPERIMENTS

It is very useful the command “SetSeed” which allows us to reply the same environment for an experiment just by setting always the same number on the display. This is important if one wants to reply more than one time an experiment in order to underline particular or different features of the model.

```
to SetSeed

  random-seed mySeed

end

to set-transparency
  ;; since turtle colors might be either numbers (NetLogo colors) or lists
  ;; (RGB or RGBA colors) make sure to handle both cases when changing the
  ;; transparency
  ifelse is-list? color
  ;; list might either have 3 or 4 member since RGB and RGBA colors
  ;; are allowed, so you can't just replace or add an item at the
  ;; end of the list.  So, we take the first 3 elements of the list
  ;; and add the alpha to the end
  [ set color lput transparency sublist color 0 3 ]
  ;; to get the RGB equivalent of a NetLogo color we
  ;; use EXTRACT-RGB and then add alpha to the end
  [ set color lput transparency extract-rgb color ]

end
```

The experiments we want to test are mainly related to the role of the Antitrust in the market, and are tested by using the sliders and the switchers, changing the parameters under control.

The first one is to test if the repressive activity of the Antitrust is useful to limit the increasing trend in profits of colluding firms above non-colluding firms. The evidence shows that there is some effect and that Antitrust helps the market to reduce illicitely gained profits which otherwise would increase linearly or exponentially.

The second experiment wants to test how the number of controls the Antitrust set up each turn can affect the profits gained by the firms and if their duration may be an important feature. The evidence shows that higher levels of controls are related to the exit of firms from the market, probably related to high level of fines. If the amount of fines and number of controls is set appropriately, it can be useful in restore a non-competitive behavior, maybe disincentiving guilty firms to collude again, since here the *collusionLevel* is fixed and firms does not learn from the past. Increasing the duration make more likely to find guilty a colluding firm and so it is useful to lead back the profits to a quasi-competitive behaviour, even though some kind of rent is always present.

The third experiment test different types of penalties: the idea here is to verify if lump sum penalties are better than penalties based on the average of profits of "normal firms" or on the procedure about the estimated dossier. Maybe the economic theory would say that lump sum fines are more effective since they do not create distortions in the behavior of the agents in the market. Unfortunately our agents are not adaptive so the distortion can not be verified. What is shown is that high level of lump sum fines lead quickly colluding firms out of the market, and in few time they all disappear. This is also due to the network effect of collusion, that is profitable when there are lots of firms colluding, but decrease in its effectiveness as long as the

number of colluding firms decreases and the Antitrust focus its attention on the few ones on the market. Fines based on the mean of profits of firms with low collusion potential try to get the outliers as firms that have profits above the average of the market and the fine is based on this computations. Since the average profit of the firms in the market is increasing and not stationary around a mean, the penalty charged on the guilty firms is increasing and it drive lots of firms out of competition. It does not seem a good way to restore market conditions.

The third method is based on the dossier created in the first day of control upon a firm. The dossier account for the first illicit profit gained by the guilty firm and when charging the penalty it will be based on this dossier, that is a number, multiplied by the duration of the controls. The problem raised is related to the possibility of under/ overestimation of the profit gained by colluding. Indeed, if for example a firm on its first day of control illicitly gets a surplus of 10, multiplied by 5 ticks of duration, will be fined for 50 even though it has earned 100 or 20. We think that this can be a good method if accurately adjusts the estimation method.

The last experiment wants to test if the incentive to cheat may be a cause of profit fluctuations. By switching on and off the *"incentive_to_cheat"* switcher it can be shown that the incentive produces changes in the slope of the line describing profit evolution over time, this is due to the effect of the cheating which restores lower prices but at the same time it increases the profits of the cheating firm. Hence sometime it increases the slope of the profit line and sometime it decreases it.

Our conclusion is that number of controls and their duration are more important in preventing collusive behavior compared to different types of penalties charged on colluding firms.

GRAPHS

Our Netlogo model uses also the *Output display* and three graphs in order to show the evolution of variables under control and to understand how the trends of the market change as market conditions are changed.

In the output are shown the initial conditions of the experiment, like number of firms in the market, duration and number of controls. The graphs can be recognized by their name and they show:

- Profits_and_prices: the average profits of firms with CollusionLevel < 3 in green and the average profits of firms with CollusionLevel > 3 in red. There are also the average price of the market which shows fluctuations and the aggregate number of firms in the market.
- Group_profits: it shows more clearly the level of profits obtained by the firms as a group, i.e. differentiating firms as far as their CollusionLevel is concerned without aggregating them under or over a threshold. Intuitively firms with high collusion potential show long run profits much higher than the others, but they are also the much more likely to exit from the market in presence of high level of penalties, displaying a break in the line somewhere in the timeline. In violet are shown firms with CollusionLevel equal to zero, in black firms with CollusionLevel equal to one, in grey firms with CollusionLevel equal to two, in orange firms with CollusionLevel equal to three and in red firms with CollusionLevel equal to four.
- Firms_number: this graph describes the same groups as before and using the same colours but describing the evolution of their number as time passes by.

XXXXX