

Cournot Competition, Organization and Learning*

Jason Barr[†] and Francesco Saraceno[‡]

October 24, 2003

Abstract

We model firms' output decisions in a repeated duopoly framework focusing on three interrelated issues: (1) the role of learning in the adjustment process toward equilibrium, (2) the role of organizational structure in the firm's decision making, and (3) the role of changing environmental conditions on learning and output decisions. We characterize the firm as a type of artificial neural network, which must estimate its optimal output decision based on signals it receives from the economic environment (which influences the demand function). Via simulation analysis we show: (1) how organizations learn to estimate the optimal output over time as a function of the environmental dynamics; (2) which networks are optimal for each environmental complexity, and (3) the equilibrium industry structure.

Keywords: *Cournot Competition, Neural Networks, Firm Learning*

JEL Classification: *C63, D83, L13, L25*

*A version of this paper was presented at the 8th International Conference of the Society for Computational Economics and Finance, Aix-En-Provence, France, June 27-29, 2002. We thank the participants, in particular Nobuyuki Hanaki and Nicolaas Vriend, for their comments and suggestions. Also, we thank Duncan Foley, Paul Sengmueller and two anonymous referees for their comments on the draft. The usual caveats apply.

[†]Department of Economics, Rutgers University Newark, Newark, NJ 07102. email: jmbarr@andromeda.rutgers.edu, Ph: 973-353-5835.

[‡]Corresponding Author: Observatoire Français des Conjonctures Économiques, 69 Quai d'Orsay, Paris 75007, France. email: francesco.saraceno@sciences-po.fr

1 Introduction

The goal of this paper is to investigate the effects of both environmental and organizational factors on the outcome of repeated Cournot games. We model the firm as an information processing network that is capable of learning a data set of environmental variables. Standard models of the firm, in general, tend to focus on the quantity strategy, while ignoring the fact that decisions are made within an organizational framework. This is particularly true for oligopolistic industries, which tend to be dominated by large firms employing thousands of workers; furthermore, these firms are run by a large group of managers that must agree on a strategy each period. These facts are often neglected in models of oligopolistic interaction, even those that focus on learning and dynamics.

Contrary to most models dealing with the dynamics of Cournot games, *we are not interested in modeling the learning of the optimal strategy per se, but rather in the learning of the economic environment.* More specifically, we model firms of different organizational structures competing, given that they have to learn the effect of changing environmental states on the demand parameters. Using this approach, we are able to investigate the relationship between optimal firm structure (in the sense of most proficient at learning the environmental characteristics) and the complexity of the environment in which quantity competition takes place. Building on a previous paper (Barr and Saraceno, 2002), we model the firm as a type of artificial neural network (ANN), which must learn to make its optimal output decision based on signals it receives from the economic environment (which influences the demand function). The use of ANNs allows us to make explicit organizational structure, and hence to include it in a model of firm competition.

We model the structure of the firm as the *size* of the network, given by the number of processing units; we show in Barr and Saraceno (2002) that firms face a trade-off between speed and accuracy. Smaller, more flexible firms learn faster, while larger firms are more accurate in the long run. In addition, we show that the solution to the problem posed by this trade-off is influenced by environmental characteristics, a position long held by management scholars. The objective of the paper is therefore to understand if, and how, this conclusion applies to the specific case of Cournot competitors facing (and having to learn) a changing demand curve, and how the complexity of environment affects optimal firm size.

The first conclusion of the paper is that neural networks are capable of

converging to the Nash equilibrium of a Cournot game. Over time, they learn to perform the mapping between environmental characteristics and optimal quantity decisions. This result is not surprising, as many adaptive algorithms have been shown to have the same property. The second –also expected– result is that profitability (linked to the proficiency of network learning) is inversely related to the complexity of the external environment and to the error firms make in trying to learn the demand parameters.

These findings constitute the background for the main results of the paper. First, given quantity competition between two firms, small firms/networks reach relatively quickly a satisfactory knowledge of the function linking environmental factors and demand; on the other hand larger firms, initially slower to learn, tend in the long run to outperform the small ones by becoming more accurate in their mapping. Related to this, we show that the optimal firm size is increasing in the complexity of the environment itself; in more complex environments the time necessary to learn the factors that determine demand is longer, so the short run competitive edge of smaller firms becomes progressively less relevant.

This result is robust, as it emerges both from a round-robin tournament between networks of different sizes, and from regression analysis on the simulation data, which shows how time, firm size, competitor’s firm size, and environmental complexity affect firm learning and hence performance. Finally, we show that an equilibrium industry configuration (in which there is no incentive to change firm size) may be found, and that it is also related to the complexity of the environment.

The paper is structured as follows. The next section briefly reviews the relevant literature, showing how we relate to (and depart from) the models on learning in oligopoly games on the one hand, and to agent-based models on the other. The following (section 3) introduces the repeated Cournot game, describing the environment and our measure of environmental complexity. Then, section 4 describes our model of the firm as a network of agents –a type of neural network– and describes its application to the duopoly example. Section 5 presents the results of the simulations and discusses the main conclusions of the paper. Finally, in section 6 we conclude with suggestions for further research and extensions.

2 Related Literature

Our work relates to two different areas. The first is the literature on Cournot competition and its dynamics. Since at least the seminal paper by Cyert and DeGroot (1973), the Cournot model has been widely studied by researchers interested in learning and strategic interaction. Some of the works in this area explore the conditions under which the duopolists will converge to the Cournot equilibrium output (recent examples include Kopel, 1996; Chiarella and Khomin, 1996; Puu, 1998; Bischi and Kopel, 2001). In these models agents have to learn how to react to their opponents' behavior. Given the different hypotheses (on demand characteristics, on externalities in the cost function, and on expectation formation), the system may be described by complex dynamics, that yield one or more equilibria; furthermore, initial conditions usually determine whether convergence occurs, or chaotic dynamics are engendered.

In a related approach, Vega-Redondo (1997), Vriend (2000) and Riechmann (2002) build on evolutionary game theory to investigate whether the Cournot outcome is stable. The last two papers, in particular, show that convergence to the Walrasian prices and quantities is more probable when social (as opposed to individual) learning takes place, and agents are boundedly rational. In these papers learning is in regards to the opponent's strategy, or the firm's own influence on prices; information about the environment (i.e., the parameters of the demand function) is either complete or unnecessary for learning to take place.

Two recent papers (Leonard and Nishimura, 1999; Bischi, Chiarella and Kopel, 2002), on the other hand, investigate the case in which duopolists lack knowledge of the demand function they face. If demand is misspecified, then even best reply dynamics may converge to steady states (pseudo equilibria) different from the unique Nash-Cournot outcome. Nevertheless, in these models there is no learning, as the misspecification is not corrected along the way.¹

This paper merges the issues described above, as we address the issues of learning *and* of misspecification. We assume that demand depends on environmental characteristics, in a way unknown to the agents. The firm observes signals from the environment and based on these signals selects an

¹Other papers (e.g., Verboven, 1997) search for conditions under which cooperation is sustainable in repeated Cournot games. Our paper does not deal with this issue.

output to produce. After observing its rival's output and the true parameters, it calculates what its output should have been, i.e., the best response. It subsequently uses the error to improve its performance, i.e., the knowledge of the relationship between observable environmental characteristics and the demand parameters. In other words, rather than learning the best strategy, firms have to learn the environment in which they operate. In this sense, as will become clear in the following pages, strategic interaction and the best response strategy remain in the background, entering the picture only as a 'benchmark' against which the firm evaluates its own performance.

As stated in the introduction, we seek to investigate how firm learning interacts with organizational features. As a consequence, the second area of the literature related to our work is that of agent-based models of the firm (Radner, 1993; Carley, 1996; DeCanio and Watkins, 1998; Li, 1999). These models, borrowing heavily from computer science, represent the firm as a network of information processing agents (nodes). In general these papers study which types of networks minimize the costs of processing and communicating information. Our model is also agent-based, as we assume that output decisions are made by an information processing network.

However, our work is different in two respects. In general, and unlike other agent-based models, we directly model the relationship between the external environmental variables, firm learning and performance; secondly, we explicitly provide an agent-based model of Cournot competition, which, to our knowledge, has not been done before. This paper is an attempt to apply a standard economic problem (Cournot competition) to a network of information processing agents to show how firms adapt to different environments in order to perform at optimal levels.

Our agent-based approach models the firm as a type of artificial neural network. ANNs are common in computer science and psychology, where they have been used for pattern recognition and modeling of the brain (Croall and Mason; 1992; Skapura, 1996). In economics, neural networks have been employed less frequently. One application –in econometrics– has been to use ANNs as non-linear estimation equations (Kuan and White, 1992). In game theory, Cho (1994) has tackled a prisoner's dilemma game using a very simple neural network (a perceptron) as a way to model bounded rationality. Because of the stochastic and non-linear nature of ANNs we employ a simulation-based approach to study the relationship between firm performance, competition and size.

3 Cournot Competition with Stochastic Demand

Suppose we have two firms competing in quantities, facing the same linear, downward-sloping demand curve, with the following profit functions:

$$\pi_i = [a - b(q_1 + q_2)] q_i - c_i q_i \quad i = 1, 2.$$

where q_i is the output decision of each firm, $c_i \geq 0$ is marginal production cost, and $a, b > 0$ are the demand parameters. Since we focus on performance, and for simplicity, we assume that costs are zero, i.e., $c_i = 0$.²

If the demand parameters were known we would have the standard scenario: each firm tries to maximize its profit, given the estimate of its rival's output, denoted $E_i q_{-i}$. The first order condition gives rise to a reaction function for each firm:

$$q_i^{br} = \frac{1}{2} \left[\frac{a}{b} - E_i q_{-i} \right],$$

where q_i^{br} is the best response output. If each player correctly assumes that the rival will produce along its reaction function then the (Nash) equilibrium output and profit for each firm are

$$q_i^* = \frac{1}{3} \frac{a}{b} \quad \pi_i^* = \frac{1}{9} \frac{a^2}{b}.$$

Textbook analysis tells us that with our linear specification of the demand function the two firms will converge to the Nash equilibrium even with backward-looking expectation formation.

3.1 The Environment

In this paper we assume that the demand parameters are stochastic in the sense that a and b are functions of environmental variables, which fluctuate according to a given probability law. For example, the intercept coefficient represents all those non-price elements that affect demand, such as preferences, income, price of substitutes, etc.; the effect of these variables on the

²Further, without any loss of generality, we assume that the firm bears no cost to carrying the network. This assumption does not affect the qualitative results of section 5.

position of the demand curve may be only known partially *ex-ante* by the firm; over time the organization has to learn how these factors indeed affect demand.

We assume that each parameter is a function of a vector of environmental states (signals), which represent the characteristics of the environment; and that each vector \mathbf{x} (of length N) belongs to the set \mathbf{X} of environmental data vectors: $\mathbf{x}_k \in \mathbf{X}$, $k = 1 \dots, v$.³ We model these environmental variables as a string of binary digits, a simple way to summarize the presence or absence of features in the environment. We refer to the current vector (indexed by time, t) $\mathbf{x}_t \in \mathbf{X}$ as to the *state of the environment*, and to \mathbf{X} as the *environment*. Each period, the state of the environment is determined by random draw from the set \mathbf{X} , where each element $\mathbf{x}_k \in \mathbf{X}$ has a fixed probability, p_k , of being selected ($p_k \geq 0$, and $\sum_{k=1}^v p_k = 1$).

Equations (1) and (2) show the functional form for the intercept and slope of the demand function:

$$a(\mathbf{x}) = c_a \left(\sum_{n=1}^N n \alpha_n x_n \right)^2, \quad (1)$$

$$b(\mathbf{x}) = c_b \sum_{n=1}^N n \beta_n x_n, \quad (2)$$

where $\alpha_n, \beta_n \in (0, 1) \forall n$ are constants, and c_a and c_b are normalizing constants so that the values of a and b are always between zero and one.⁴ This characterization says for example that the n^{th} bit for the slope has a marginal contribution of $n \beta_n \Delta x_n$. By multiplying each bit by its index n we sort them in order of importance, x_1 being the element that contributes the least, and x_N being the most important. Notice furthermore that, though not necessary, we assume this order of importance to be the same for a and b .

³ \mathbf{X} is a subset of the set of all binary digit vectors of length $N (= 25)$, which has 2^N elements. In the simulations below, we pick $\nu (= 25)$ vectors by random draws. This data set remains constant throughout all the simulations (generating alternative data sets does not affect the qualitative results presented in section 5).

⁴We add the square term on the intercept in order to increase the difficulty of the learning problem, since our interest is creating a model where many agents are needed in order to learn.

3.2 Complexity

In our model competition occurs in two areas: along the reaction curve (i.e., one firm's increased output affects, via the price, the other firm's profit) and along the 'learning dimension.' That is to say, the better and/or faster a firm is able to estimate the demand parameters, the more it has a competitive advantage, in the sense that it will have a relatively higher profit compared to its rival. We discuss this type of advantage in section 4.2.

Firms have to learn to recognize how environmental changes will affect demand and hence their optimal output. In other words, they have to learn how to map the observed environmental vector \mathbf{x} into the values of a and b . We define the complexity of this pattern recognition problem as the *entropy* of the probability distribution generating the environmental data points. Heuristically, we can think of entropy as a measure of the *quantity of information* the firm is likely to process. If the distribution is concentrated on one or two points, for example, then it is very likely to see those points most often; while in a more uniform distribution the firm is more likely to see all the different states. In other words, entropy is a measure of 'disorder,' and as such we take it as a measure of complexity. Given the probability distribution associated with the set \mathbf{X} , the entropy is defined as

$$E = - \sum_{k=1}^v p_k \ln(p_k).$$

Entropy ranges between 0 for a degenerate distribution and $\ln(v)$ for a uniform distribution.⁵

4 The Firm as a Network of Information Processing Agents

In the previous section we argued that competition between firms occurs in an unknown environment; firms must learn to map observed environmental characteristics to unknown demand parameters. For this reason we view the firm as an *information processing algorithm*. As mentioned in section 2, we model the firm as a network of information processing agents. In Barr

⁵In the simulations below complexity classes are generated by random draws. Since we have $p_k > 0, \forall k$ the minimum entropy for our data set is strictly greater than zero.

and Saraceno (2002) we highlight the features of firm behavior discussed by management scholars (such as Galbraith, 1973 and Lawrence and Lorsch, 1986) that can be modeled by ANNs: firms/organizations *process information in a decentralized manner*, both *serially and in parallel* (i.e., information within hierarchical levels is processed simultaneously, while it is processed serially between levels). Organizations *learn by experience* and they learn to *generalize* their experience to other related situations; this learning involves both costs and benefits, for which there is an optimal firm size. Further, the knowledge of the firm does not reside in any one agent but rather *resides in the network of agents*. Finally, *firms are capable of adapting to their environment*.

We model the firm as a type of Backward Propagation Network (BPN) (Skapura, 1996). A graphical representation of the network is shown in Figure 1. The network has three layers. The data (information) layer is comprised of signals from the environment. As mentioned above, a particular data vector, \mathbf{x} , is a string of binary digits which represents whether features from the environment are absent (0) or present (1). A 'hidden' (management) layer is comprised of several processing units (nodes), and the 'output' (CEO) layer is comprised of a single, final processing unit. Processing within a layer occurs in parallel; processing between layers occurs serially. Each node performs the same action: it takes a weighted sum of the inputs and then applies a squashing (sigmoid) function which outputs values between 0 and 1 (i.e., large negative values are squashed to 0, large positive values are squashed to 1, and intermediate values are assigned a value close to 0.5).⁶

[Insert Figure 1 here]

Furthermore the network is capable of learning a data set, i.e., the economic environment which affects demand. Over successive iterations (estimations), as the network processes information, it improves its relative performance, i.e., it learns to more accurately estimate the mapping between environmental characteristics and demand parameters. As shown in Barr and Saraceno (2002), ANNs highlight the trade-offs to firm learning: small

⁶A possible interpretation of the network is as a group of agents (managers) who assign values to the environmental signals and pass these values up the hierarchy to a CEO, who then takes an output decision based on these values. The CEO then observes the true best-response output and communicates this information down the hierarchy to the managers who use this information to improve their performance in the future periods.

firms learn faster, but less precisely, while larger firms are slower to adapt but are better able to learn a in complex environment.

4.1 The Network

Here we discuss the workings of the network in more detail. As we mentioned, the environmental data (information) layer is a binary vector $\mathbf{x} \in \mathbf{X}$ of length N . Each of the M_i nodes (managers) in the hidden (management) layer takes a weighted sum of the data from the data layer. That is, the j^{th} agent in the hidden layer of firm i calculates

$$y_{ij}^h = \mathbf{w}_{ij}^h \mathbf{x} \equiv [w_{ij1}^h x_1 + \dots + w_{ijn}^h x_n + \dots + w_{ijN}^h x_N],$$

where $i = 1, 2, j = 1, \dots, M_i, n = 1, \dots, N$, and $w_{ijn}^h \in \mathbb{R}$ (time subscripts removed for notational convenience). Thus the set of 'inputs' to the M_i agents of the management/hidden layer is

$$\mathbf{y}_i^h = (y_{i1}^h, \dots, y_{ij}^h, \dots, y_{iM_i}^h) = (\mathbf{w}_{i1}^h \mathbf{x}, \dots, \mathbf{w}_{ij}^h \mathbf{x}, \dots, \mathbf{w}_{iM_i}^h \mathbf{x}).$$

Each agent then transforms the inputs via a sigmoid (voting) function to produce an output, $z_{ij}^h = g(y_{ij}^h) = 1/(1 + e^{-y_{ij}^h})$. The vector of processed outputs from the hidden layer is

$$\mathbf{z}_i^h = (z_{i1}^h, \dots, z_{ij}^h, \dots, z_{iM_i}^h) = (g(y_{i1}^h), \dots, g(y_{ij}^h), \dots, g(y_{iM_i}^h)).$$

The inputs to the output (CEO) layer is a weighted sum of all the outputs from the hidden layer:

$$y_i^o = \mathbf{w}_i^o \mathbf{z}_i^h \equiv (w_{i1}^o z_{i1}^h + \dots + w_{ij}^o z_{ij}^h + \dots + w_{iM_i}^o z_{iM_i}^h)$$

where $w_{ij}^o \in \mathbb{R}$. Finally, the output of the network –the estimate of the quantity \hat{q}_i – is determined by transforming y_i^o via the sigmoid function, $\hat{q}_i = g(y_i^o)$. We summarize the data processing ('feed-forward') phase of a network with one hidden layer as⁷

$$\hat{q}_i = g \left[\sum_{j=1}^{M_i} w_{ij}^o g(\mathbf{w}_{ij}^h \mathbf{x}) \right] \equiv NN_i(\mathbf{x}). \quad (3)$$

⁷We focus on a one hidden layer network to simplify the computation and to reduce the number of organizational variables. Increasing complexity (size) of the firm might also be modeled by adding hidden layers rather than (or in addition to) nodes. The use of additional layers would be justified only if we had a very complex learning problem, which is not the case for our simple model.

Notice that the expected value of the opponent's quantity decision doesn't directly enter into equation (3) (that is, we don't have $\hat{q}_{it} = NN(\mathbf{x}_t, E\hat{q}_{-it})$). In fact, if this expectation were adaptive (i.e., it included past observations of q_{-i}) then it would be unwarranted since the state of the environment changes from period to period, making past quantity decisions moot. If the expectation were not adaptive, it would have to be based on the information available to the firm, i.e., \mathbf{x} . This would bring us back to equation (3).

Also notice that the lack of direct strategic interaction does not imply that the two firms do not influence each other. In fact, the competitor's choice enters into the best response 'ideal' quantity of the firm, and consequently affects the weight update process described in section 4.3 below. In this framework, firm $-i$'s actions affects firm i 's payoffs, rather than firm i 's actions. In addition, firms will be able to learn as long as the behavior of the rival is not too erratic, though the rate of learning will be affected.⁸

In the next section we will show that the firm maximizes profit by minimizing the error it makes in choosing a quantity to produce; then, in section 4.3 we'll describe how learning takes place.

4.2 Learning and Profitability

Each period firms observe an environmental state vector, \mathbf{x} , and produce an output given by:

$$\hat{q}_i = NN_i(\mathbf{x}), \quad i = 1, 2$$

Given the output choices of the firms, the profit function for firm i is given by

$$\pi_i = [a - b(\hat{q}_1 + \hat{q}_2)] \hat{q}_i.$$

Each firm then compares its output choice to the (optimal) quantity that it would have chosen if it produced along its reaction curve, given its rival's choice of output, \hat{q}_{-i} (see footnote 11 for further discussion on the true demand parameters):

$$q_i^{br} = \frac{1}{2} \frac{a}{b} - \frac{1}{2} \hat{q}_{-i}.$$

⁸We ran some tests, and the results show that only in extreme cases, when the competitor, for example, chooses a random quantity with very high variance, the firm may be unable to learn the mapping from environmental characteristics to demand parameters.

Given the rival's choice of output, \hat{q}_{-i} , the highest profit firm i could have achieved if it produced its optimal output, q_i^{br} , is given by

$$\pi_i^{br} = \left[a - b \left(\frac{1}{2} \frac{a}{b} - \frac{1}{2} \hat{q}_{-i} + \hat{q}_{-i} \right) \right] \left[\frac{1}{2} \frac{a}{b} - \frac{1}{2} \hat{q}_{-i} \right] = b (q_i^{br})^2.$$

Next we define the *loss*, L_i , as

$$L_i = (\pi_i^{br} - \pi_i) = b \left[(q_i^{br})^2 + \hat{q}_i^2 - 2\hat{q}_i q_i^{br} \right] = b (q_i^{br} - \hat{q}_i)^2. \quad (4)$$

Thus we can define the estimation error that the firm makes each period as

$$\xi_i = (q_i^{br} - \hat{q}_i)^2, \quad (5)$$

and per-period profit of firm i can be given by

$$\pi_i = \pi_i^{br} - L_i = \pi_i^{br} - b\xi_i.$$

Since firm i 's profit is maximized when $\xi_i = 0$, the firm attempts to minimize ξ_i over time, which it does via the learning algorithm procedure described in the next section.

4.3 The Learning Algorithm

In section 4.1, we described how firms choose a quantity, \hat{q}_i , given an observed state of the environment. Over time, however, firms improve their performance as they learn to map different states to the correct demand parameters. Each period \hat{q}_i is compared to the ideal output i.e., the output along the reaction function and the error is calculated using equation (5). This information is then propagated backwards as the weights are adjusted according to the learning algorithm, which aims to minimize the squared error, ξ_i .

Recall that for the sigmoid function $g'(y_i^o) = \hat{q}_i(1 - \hat{q}_i)$, and define $z_{ij}^h = \partial(y_i^o) / \partial(w_{ij}^o)$. We can write the gradient of ξ_i with respect to the output-layer weights as

$$\frac{\partial \xi_i}{\partial w_{ij}^o} = - (q_i^{br} - \hat{q}_i) \hat{q}_i (1 - \hat{q}_i) z_{ij}^h,$$

Similarly, we can find the gradient of the error surface with respect to the hidden layer weights:

$$\frac{\partial \xi_i}{\partial w_{ijn}^h} = -z_{ij}^h (1 - z_{ij}^h) [(q_i^{br} - \hat{q}_i) \hat{q}_i (1 - \hat{q}_i)] w_{ij}^o x_n.$$

The weights are then adjusted a small amount in the opposite (negative) direction of the gradient. A constant, η , is the learning-rate parameter which smooths the updating process ($\eta = 10$ in the simulations below). Thus if we define $\delta_i^o = (q_i^{br} - \hat{q}_i) \hat{q}_i (1 - \hat{q}_i)$, the weight adjustment for the output layer is

$$w_{ij}^o(t+1) = w_{ij}^o(t) + \eta \delta_i^o z_{ij}^h.$$

Similarly, for the hidden layer,

$$w_{ijn}^h(t+1) = w_{ijn}^h(t) + \eta \delta_{ij}^o x_n,$$

where $\delta_{ij}^o = z_{ij}^h (1 - z_{ij}^h) \delta_i^o w_{ij}^o$. When the updating of weights is finished, the firm views the next input pattern and repeats the weight-update process.⁹

5 Learning and Cournot Competition: A Simulation Experiment¹⁰

As a summary, let us review the estimation and learning steps for a given entropy value:

1. The firm $i = 1, 2$ observes the state of the environment $\mathbf{x}_t \in X$.
2. Based on the observation of \mathbf{x}_t , it estimates how much to produce: $\hat{q}_{it} = g \left[\sum_{j=1}^{M_i} w_{ij}^o(t) g(\mathbf{w}_{ij}^h(t) \mathbf{x}_t) \right]$.

⁹We begin with a completely untrained network by selecting random weight values (i.e., we assume the network begins with no prior knowledge of the environment).

¹⁰The simulations in the section were performed in *Mathematica* 3.0. The code is available upon request.

3. It then observes the true parameters and calculates the optimal quantity along the reaction function $q_{it}^{br} = \frac{1}{2} \left[\frac{a(\mathbf{x}_t)}{b(\mathbf{x}_t)} - \hat{q}_{-it} \right]$.¹¹
4. The difference between \hat{q}_{it} and the best response q_{it}^{br} serves as basis for the weight update process: $\xi_{it} = (q_{it}^{br} - \hat{q}_{it})^2$. Using this error, the firm updates its weights to improve its performance in the next round.
5. The price determined by the market is $p_t = [a - b(\hat{q}_{1t} + \hat{q}_{2t})]$, and based on that we calculate actual profit. Profit and the error are recorded at each t . The steps are repeated again, until we reach $t = T$.

We ran each of the experiments described below 50 times and took averages in order to smooth fluctuations due to the random initial weights value.

5.1 Experiment #1: Learning Optimal Quantity

This section shows that networks converge to the optimal quantity. The weight update process allows the network to make a correct mapping between environmental characteristics and optimal quantity. To keep things simple, for the moment we consider two firms of equal size.¹²

Figure 2 shows how firm 1 converges to the optimal quantity. The curves represent the ratio of profit to the optimal value ($\pi_{1,t}/\pi_t^*$), and the difference between quantity and optimal quantity ($|q_{1t} - q_t^*|$). Convergence implies that the first curve goes to one, whereas the second converges to zero. This is in fact what happens. Profits and quantity tend to their Cournot-Nash optimal values, π_t^* and q_t^*

[Insert Figure 2 here]

The second goal of this section is to show that increasing complexity of the environment has, in general, the effect of reducing the network performance.

¹¹With a linear demand function, firms have to observe two pairs of $(p, q_1 + q_2)$ to determine the true parameters a and b (and to compute the best response). Hence, we implicitly assume that for each draw of the environmental vector firms play at least twice, i.e., that the environment remains constant sufficiently long. Since we focus on the learning of the mapping from \mathbf{x} to the parameters, and within each iteration no new information is provided, these ‘subperiods’ can be neglected in the analysis.

¹²In particular, $M_1 = M_2 = 8$. In this initial experiment the choice of network size is not crucial. In general, networks of different sizes will converge at different speeds, but none will fail to learn the optimal quantity.

Figure 3 plots the average profit over the $T = 200$ iterations, against the entropy value used for that particular run. As is expected, the average π/π^* drops for increasing complexity; we also plot the deviation from optimal quantity, which increases with entropy.¹³

[Insert Figure 3 here]

In this section we showed that networks are able to learn the optimal strategy in a Cournot setting, and that this learning is easier the simpler is the environment they face. Given these general features of our framework, we'll turn to the main topic of this paper: the relative performance of networks of different organizational structures competing in environments of varying complexity.

5.2 Experiment #2: Optimal Network Size

Tournament

This section investigates the performance of firms of different sizes competing against each other. We designed the experiment as a round robin between networks with 2 to 15 nodes in the hidden layer, so that we had $\binom{14}{2} = 91$ games. We divided the environment in two different levels of complexity, depending on the entropy value: simple environments have an entropy going from 1.4 to 1.9, whereas complex ones have an entropy going from 2.7 to 3.2.¹⁴ We had 50 draws of entropy values within each class (simple/complex), and we recorded the average profit and error for the two firms over the $T = 200$ iterations. The total score of each network is the sum of the average profits obtained against all the other opponents.¹⁵

The results are reported in figure 4. The winners in the simple environment tournament are networks of size 3; whereas in the complex case the highest scoring networks have 7 nodes. Furthermore, looking at the extremes, we see that small networks perform quite well in the simple environment whereas the larger ones are relatively more effective in the complex environment. Finally, notice that profits in the simple environment have higher

¹³In the early stages of the process profits may be negative. We overlook this issue by assuming that firms have enough internal funds to cover initial losses.

¹⁴These values are obtained by dividing the total range of E in three intervals of equal length, and discarding (for this section) the middle one.

¹⁵This measure, which avoids the distortions linked to measures of relative profit, was suggested by Nicolaas Vriend.

mean and standard error (0.311 and 0.0048 respectively, against 0.199 and 0.0042 for the complex environment).

[Insert Figure 4 here]

To summarize, the tournament tells us that computational power may be a disadvantage when the environment is simple. In this case smaller firms, which converge more rapidly, and have an overall better performance. Since this velocity is paid in terms of lower accuracy, in complex environments the profitability is reversed, and the higher accuracy of large firms is rewarded, compensating for slower convergence.

Regression Analysis

The results of the tournament are confirmed by another experiment we ran. We made 2000 random draws of the parameters $E \in [1.4, 3.2]$, $M_1, M_2 \in [2, 15]$, and $T \in [30, 300]$. We then made the two networks compete, and recorded the average profits and squared errors over each run. Finally, we ran a regression using as the dependent variable average profits. Table 1 reports the results for firm one (given the symmetry, results for firm two are analogous).

Notice that, over the relevant range, entropy and the number of iterations have the expected signs. Namely, an increasing complexity (higher E) yields lower average profit; and an increase in the number of runs T gives to the firms more time to learn, causing lower average error and higher average profit, all else equal. A more interesting relationship exists between the number of nodes and our measures of firm performance. Figure 5 plots the profit of firm one using the coefficients of the relevant variables, M_1 and M_2 .

[Insert Figure 5 here]

The figure shows that firm one's profit is increasing in the other firm's dimension, even if at a decreasing rate. We will come back to this relationship when discussing optimal size in experiment #3.

The relationship with the firm's own dimension is not as clear from the picture, but it suggests the hump-shape relationship we highlighted before. We therefore investigated the relationship between a firm's size and complexity. Figure 6 shows how profits depend on firm one's size and entropy, both in the relevant ranges. It shows that profit is decreasing in complexity, whereas with respect to the firm size the bell shape is clearer for higher entropy levels.

Dep. Var.: $\sum \pi_1/T$					
Variable	Coefficient	t-stat	Variable	Coefficient	t-stat
<i>const</i>	111.02	34.7	M_2	8.37	30.8
E	5.71	2.6	M_2^2	-0.356	-12.3
E^2	-2.59	-6.0	M_2^3	0.00794	7.5
T	1.52	41.1	$M_1 \cdot M_2$	-0.0986	-8.5
T^2	-0.01	-26.1	$(M_1 \cdot M_2)^2$	0.000138	4.0
T^3	0.0000337	20.0	$(E \cdot M_1)^2$	-0.00445	-11.6
T^4	-0.00000004	-16.7	$T \cdot M_1$	0.00223	3.4
M_1	1.307	4.8	$T \cdot M_2$	-0.0203	-30.5
M_1^2	-0.136	-4.7	$(T \cdot M_2)^2$	0.000002	19.2
M_1^3	0.00349	3.3			
R^2	0.959				
<i>obs</i>	2000				

Table 1: Regression of profits over E , T , M_1 and M_2 , including powers and cross terms. All coefficients are multiplied by 10^4 to make the table more readable. Non significant variables were omitted.

This explains why in the previous figure, when entropy was held constant, the relationship was less visible.

[Insert Figure 6 here]

To sum up, the regression results confirm the findings of the tournament, and add new insights. Increasing complexity, and shorter time to learn, negatively affect profits, whereas the hump-shape relationship between own size and profitability is more evident at higher complexity levels. We also found that profits are linked (positively) to the opponent's dimension, an issue that we'll tackle next.

5.3 Experiment #3: Firm Size Equilibria

As discussed above, firms learn to produce at the Nash equilibrium output level over time. Here, however, we explore the concept of equilibrium with regards to network size, which we call a network size equilibrium (NSE). We define an NSE as a pair of M 's such that neither firm has an incentive to change the number of managers. That is to say, in an equilibrium, each

network, given the number of agents (nodes) of its rival, finds that switching to another number of agents will decrease its average profit. Thus, the equilibrium is a pair $\{M_1^*, M_2^*\}$ such that

$$\Pi_i(M_i^*, M_{-i}^*) \geq \Pi_i(M_i, M_{-i}^*), \quad \forall M_i \quad i = 1, 2.$$

where

$$\Pi_i = \frac{1}{T} \sum_{t=1}^T \pi_{it}(M_i, M_{-i}).$$

We ask the questions: does (at least one) NSE exist for each entropy? And what is the relationship between complexity and the equilibrium size networks? We focus on the equilibria that exist after T periods, and do not have endogenous dynamics with the number of managers M_i , i.e., we do not examine firms changing the number of managers during the learning process. Rather we conduct a kind of comparative statics exercise, whereby we look at the NSE that arise for given environmental conditions.

In this experiment, we have networks of different sizes compete for $T = 200$ iterations (for each pair of M 's and entropy value we take 50 runs and take averages to smooth out fluctuations in each run). That is to say, firm one and two compete against each other for each size, $M_1, M_2 \in \{2, \dots, 15\}$. We repeat this competition for several different entropy levels. We then look to see, for each entropy value, if one (or more) NSE exist.

As an example, figure 7 shows the profits of the two firms, when holding constant firm two's network size at six nodes, and increasing the size of firm one. We see that firm one's optimal size given firm two's size (six nodes) is at five nodes. We also see that after that value, there is a negative relationship between firm one and two's profits. As we increase firm one's nodes above a certain value, it achieves a tremendous competitive disadvantage *vis a vis* firm two because its network is much larger and slower in converging to the correct weight values.

[Insert Figure 7 here]

In the following graph we present the results of the simulation. We generated 30 different entropy values. For each of them we calculated the NSE and then added the total number of managers to obtain an 'equilibrium industry size.' This gave us a data set of 62 NSE's and industry sizes, for which there was at least one equilibrium.

We grouped each of the NSE's into three entropy/complexity categories: simple, medium and complex, where the ranges are the same as in section 5.2. We then took the average industry size for each category. Figure 8 shows the results. As we can see average industry size is increasing in complexity. This concords with the findings discussed in experiment #1 and #2.

[Insert Figure 8 here]

6 Conclusion

This paper offers some insights on the interaction between the competition in an oligopolistic market and learning performance linked to organizational structure and environmental complexity. In general, we showed that neural networks as models of the firm are able to converge to the unique Nash equilibrium of a Cournot game when facing a linear demand function with stochastic parameters.

We also showed that the optimal organizational structure is not constant, but changes with the environment. The trade-off, that we investigated in more general terms in a previous paper, appears in a Cournot setting as well: speed and accuracy are inversely related, and which factor is more profitable depends on, in general, the complexity of the environment. This result emerges from different types of experiment: a round robin tournament, a regression analysis, and an investigation of the equilibrium structure of the industry.

A number of open questions remain for further investigation in future research. One is whether this structure can give us insights on the incentives for collusive behavior, and on their relationship with the environment. We could, for example, investigate the relationship between environmental complexity, firm complexity and cooperative behavior. This structure could also be useful to investigate quantity output equilibrium selection (a case that did not concern us here, given that we dealt with linear demand, and consequently with a unique equilibrium). Finally we could explore the general analytical properties of ANNs as they relate to economic phenomena, such as those explored here.

References

- Barr, J. and F. Saraceno (2002), "A Computational Theory of the Firm," *Journal of Economic Behavior and Organization*, 49, 345-361.
- Bischi, G., C. Chiarella and M. Kopel (2002), "On Market Games with Mismatched Demand Functions: Long Run Outcomes and Global Dynamics," Mimeo, University of Urbino, University of Technology, Sidney, University of Technology Vienna, September 2002.
- Bischi, G. and M. Kopel (2001), "Equilibrium Selection in an Nonlinear Duopoly Game with Adaptive Expectations," *Journal of Economic Behavior and Organization* 46, 73-100.
- Carley, K. M. (1996), "A Comparison of Artificial and Human Organizations." *Journal of Economic Behavior and Organization*, 31, 175-191.
- Chiarella, C. and A. Khomin, (1996), "An Analysis of the Complex Dynamics Behavior of Nonlinear Oligopoly Models with Time Delays," *Chaos, Soliton & Fractals*, 7(12), 2049-2065.
- Cho, I.-K. (1994), "Bounded Rationality, Neural Networks and Folk Theorem in Repeated Games with Discounting," *Economic Theory*, 4, 935-957.
- Croall, I. F. and J. P. Mason (1992), *Industrial Applications of Neural Networks: Project ANNIE Handbook*, Springer-Verlag, New York.
- Cyert, R.M. and M.H. DeGroot (1973), An Analysis of Cooperation and Learning in a Duopoly Context, *American Economic Review*, 63, 24-37.
- DeCanio, S. J. and W. E. Watkins (1998), "Information Processing and Organizational Structure," *Journal of Economic Behavior and Organization*, 36, 275-294.
- Galbraith, J., (1973). *Designing Complex Organizations*. Addison-Wesley Publishing, Reading.
- Kopel, M. (1996), "Simple and Complex Adjustment Dynamics in Cournot Duopoly" *Chaos, Soliton & Fractals*, 7, 2031-2048.
- Lawrence, P. R. and J. W. Lorsch (1986), *Organization and Environment*. Harvard Business School Press, Boston.

- Léonard, D. and K. Nishimura (1999), “Nonlinear Dynamics in the Cournot Model Without Full Information,” *Annals of Operations Research*, 89,165–173
- Li, H. (1999), “Hierarchies and Information-Processing Organizations,” *Review of Economic Design*, 4, 1010-126.
- Kuan, C.-M. and H. White, (1994), “Artificial Neural Networks: An Econometric Perspective” *Econometric Reviews*, 13, 1-91.
- Riechmann, T. (2002), “Cournot or Walras? Agent Based Learning, Rationality, and Long Run Results in Oligopoly Games,” *Universität Hannover, Diskussionspapier Nr. 261*, August 2002.
- Puu, T. (1998), “The Chaotic Duopolists Revisited,” *Journal of Economic Behavior and Organization*, 33, 385-394.
- Radner, R. (1993), “The Organization of Decentralized Information Processing,” *Econometrica*, 61, 1109-1146.
- Skapura, D. M. (1996), *Building Neural Networks*, Addison-Wesley, New York.
- Vega-Redondo, F. (1997), “The Evolution of Walrasian Behavior,” *Econometrica*, 65, 375-384.
- Verboven, F. (1997), “Collusive Behavior with Heterogeneous Firms,” *Journal of Economic Behavior and Organization*, 33, 122-139.
- Vriend, N. J. (2000), “An Illustration of the Essential Difference Between Individual and Social Learning, and its Consequences for Computational Analyses,” *Journal of Economic Dynamics and Control*, 24, 1–19.

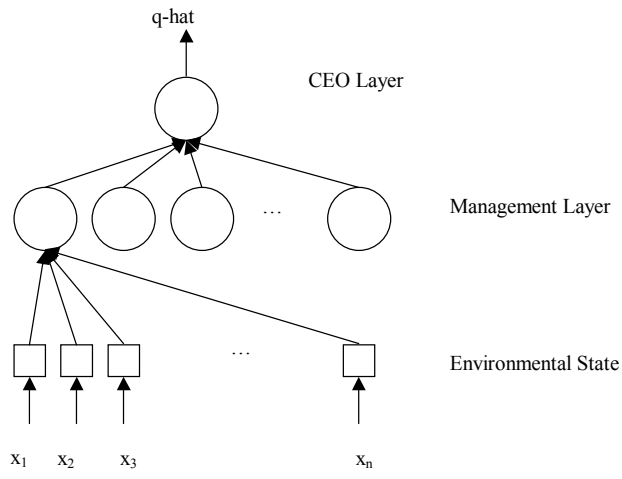


Figure 1: Network of Managers

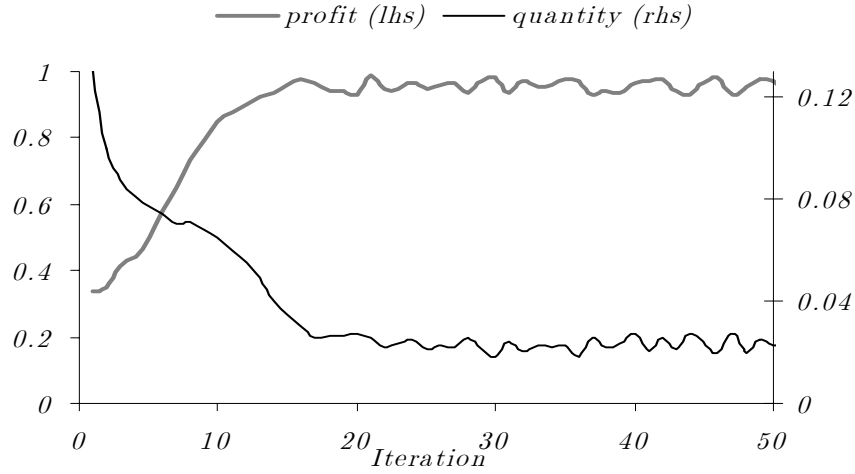


Figure 2: Convergence to Cournot Nash Equilibrium for one of the two competitors. Ratio of profit to optimal value (π_{1t}/π_t^*), and absolute value of the difference between quantity and optimal quantity ($|q_{1t} - q_t^*|$). $T = 200$ (only the first 50 iterations are shown).

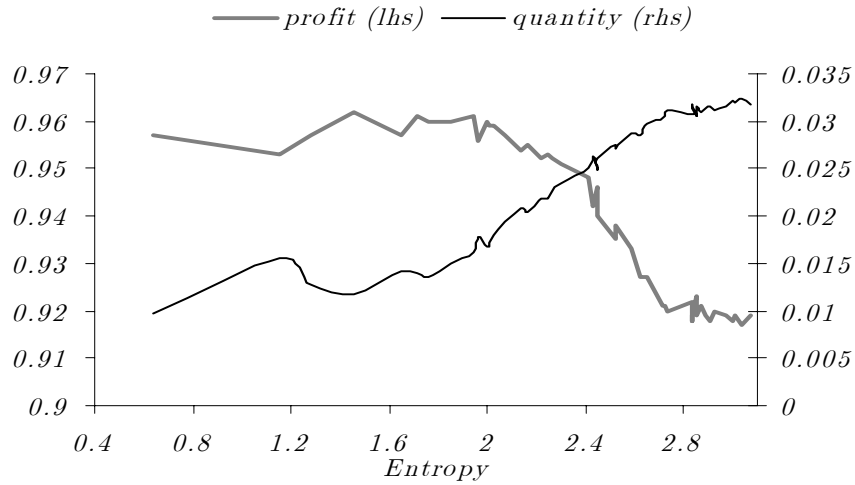


Figure 3: Complexity and network performance. Average profit ratio, $T^{-1} \sum_t (\pi_{1t}/\pi_t^*)$, and quantity deviation, $T^{-1} \sum_t (|q_{1t} - q_t^*|)$, for increasing entropy values.

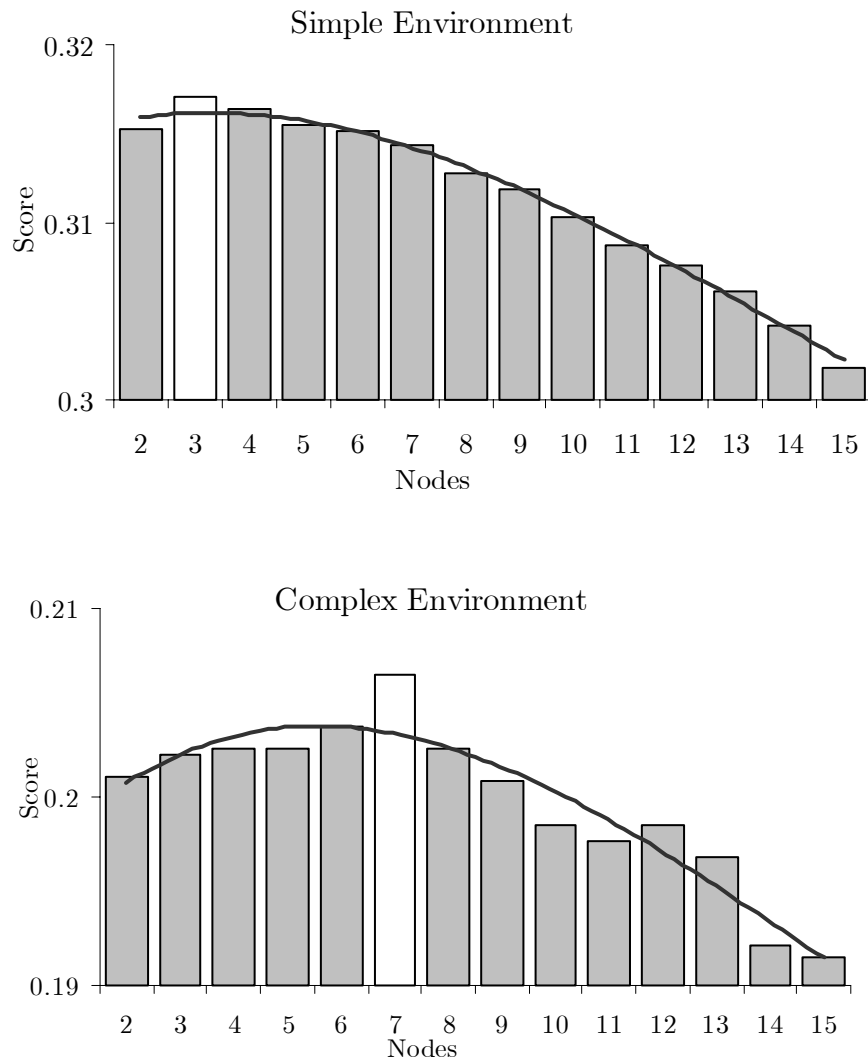


Figure 4: Round Robin Tournament. Scores for simple and complex environments. Third order approximating polynomials are also plotted.

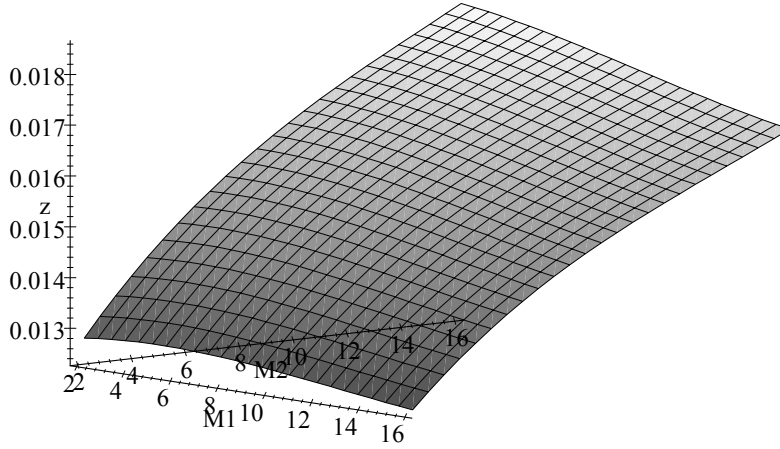


Figure 5: Average profit of firm 1 as a function of size of the two firms:
 $10^4 \cdot \bar{\pi}_1 = 111 + 1.3M_1 - 0.13M_1^2 + 0.003M_1^3 + 8.3M_2 - 0.3M_2^2 + 0.007M_2^3 - 0.09M_1M_2 + 0.0001(M_1M_2)^2$

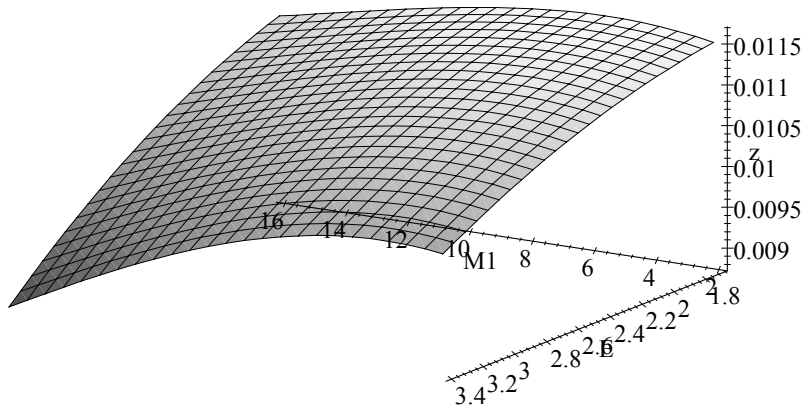


Figure 6: Average profit of firm 1 as a function of own size and entropy:
 $10^4 \cdot \bar{\pi}_1 = 111 + 1.3M_1 - 0.13M_1^2 + 0.003M_1^3 + 5.7E - 2.5E^2 + -0.004(E \cdot M_1)^2$

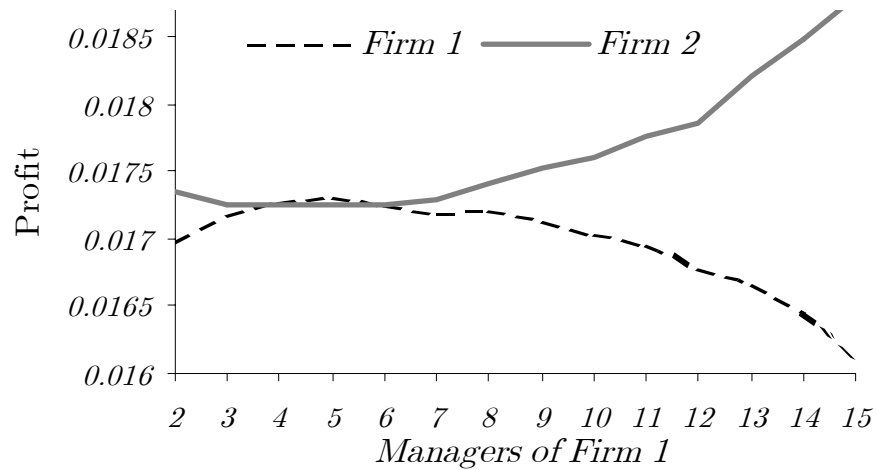


Figure 7: Profits when $M_2 = 6$ and M_1 changes between 2 and 15. ($E = 1.45$).

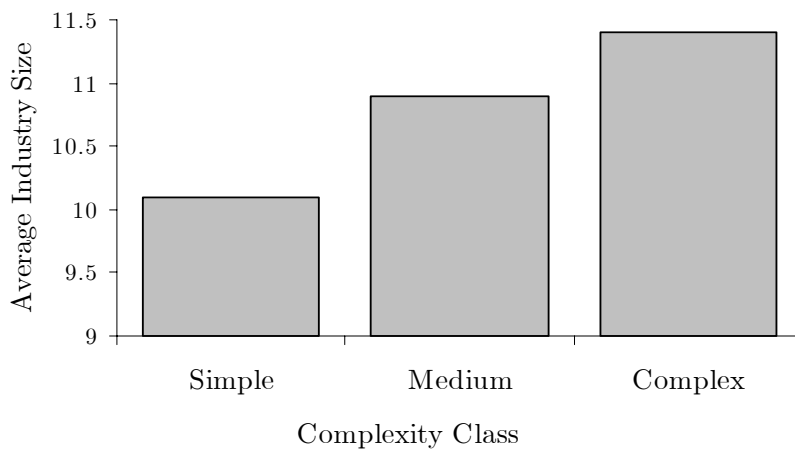


Figure 8: Average industry size ($M_1 + M_2$) vs. environmental complexity