MDPI

Article

Deep Learning Pricing of Processing Firms in Agricultural Markets

Hamed Khalili

Research Group E-Government, Faculty of Computer Science, University of Koblenz, D-56070 Koblenz, Germany; hamedkhalili@uni-koblenz.de

Abstract: The pricing behavior of agricultural processing firms in input markets has large impacts on farmers' and processors' prosperity as well as the overall market structure. Despite analytical approaches to food processors' pricing in agricultural input markets, the need for models to represent complex market features is urgent. Agent-based models (ABMs) serve as computational laboratories to understand complex markets emerging from autonomously interacting agents. Yet, individual agents within ABMs must be equipped with intelligent learning algorithms. In this paper, we propose supervised and unsupervised learning agents to simulate the pricing behavior of firms in agricultural markets' ABMs. Supervised learning firms are pre-trained to accurately best respond to their competitors and are deemed to result in the market Nash equilibria. Unsupervised learning firms play a course of pricing interaction with their competitors without any pre-knowledge but based on deep reinforcement learning. The simulation results show that unsupervised deep learning firms are capable of approximating the pricing equilibria obtained by the supervised firms in different spatial market settings. Optimal discriminatory and uniform delivery pricing emerges in agricultural input markets with the high and intermediary importance placed on space. Free on board pricing emerges in agricultural input markets with small importance placed on space.

Keywords: agent-based model; oligopsony; duopsony; spatial competition; spatial price theory; game theory; deep learning; price discrimination; free on board pricing; uniform delivery pricing

1. Introduction

Although microeconomic textbooks often introduce agricultural markets as examples for perfectly competitive markets, a large number of studies have shown that such markets often indicate features of imperfect competition [1,2]. In food processing input markets, a large number of spatially dispersed farms supply the primary input to a small number of large processing firms. The spatial pricing of processing firms in input markets—competing for farmers' products—is investigated in the existing literature, often based on the models of oligopsonistic competition [3–10]. The food processors' spatial competition has short- to long-term impacts on emerging prices in procurement markets. This determines the prosperity of farmers and processors as well as the overall structure of the market.

Most existing theoretical approaches in the literature [11–14] utilize analytical methods to understand the procurement pricing policies of processing firms. Although these approaches develop solid mathematical models to explain the emerging prices, they are not suited to describing pricing policies in the real complex market environments:

- Whereas these models often presume only two extreme pricing policies, namely free on board (FOB) pricing (where the processors set the farm gate price and farms must pay the total transportation cost from farm gate to the processing company gate) and uniform delivered (UD) pricing (where the processors set the farm gate price and bear the entire transportation costs), in the real life markets, the processing firms are free to choose prices with various possible degrees of absorbing transport costs comprising

not only the FOB to UD but also in-between degrees of shared transport costs to be absorbed by both the purchasing firms and the farmers.

- Whereas these models often assume that the interactions of firms takes place in one-stage games, real life markets can incorporate infinitely dynamic firm interactions.

Agent-based models ABMs [15] have recently been proposed approaches to cope with the deficiencies mentioned above. These frameworks are utilized to facilitate the simulation of the interaction of autonomous agents in complex environments [16]. Yet, individual agents within ABMs must be equipped with appropriate decision-making mechanisms in order to enable the entire model to successfully simulate emergent behavior at the system level [17]. Thus, agents must discover a solution autonomously through observations and relying on their own knowledge by means of *learning* [18].

Our objective in this paper is to develop algorithms based on artificial intelligence (AI) to overcome the learning problem within ABMs of agricultural input markets. The main research questions (RQs) driving this research are:

RQ 1: How to build AI-based pricing firms which autonomously choose optimal prices in the food processing input market given the competitors' policies?

RQ 2: How to design a proper game theoretical benchmark to assess the decency of the learned prices by learning firms?

RQ 3: Which pricing policies emerge as equilibrium in agricultural input markets under different spatial market settings?

To answer these RQs, we constitute two types of agents: agents who we name *supervised* agents throughout and agents who we name *unsupervised* agents throughout. Supervised agents only confront supervised agents through our study and the unsupervised agents only confront unsupervised agents through our study. The unsupervised agents are deep learning agents, which are created to answer RQ 1. The supervised agents are created in our paper to answer RQ 2. While the unsupervised agents are involved in a course of interactive learning with their competitors, without any pre-knowledge, the supervised agents are pre-trained with a complete list of best response policies given the price of the opponent (which is computationally prepared before the market interactions begin). As the sequential process of mutual best responding will lead the system towards the same or similar termination points as the Nash equilibrium [19], the supervised agents are thought to be the benchmark in our study. Hence, we use the supervised agents' pricing outcomes as reliable criteria to evaluate the performance of the unsupervised agents with respect to achieving (or not achieving) the targeted game Nash equilibria prices under different spatial market settings. Finally, based on the results of market simulations with regard to the equilibrium behavior of supervised agents and unsupervised agents under different spatial market settings (resulted from varying the global transportation cost rate and varying the price elasticity of supply), we answer RQ 3.

The remainder of the paper is organized as follows. Section 2 presents the literature background with regard to pricing theories in agricultural input markets as well as learning theories within multi-agent systems. Section 3 presents our ABM's underlying spatial market together with the processors' and farms' characteristics. Section 4 conveys the algorithms, which the supervised agents and unsupervised agents apply, respectively. The results of the simulations of supervised and unsupervised agents by varying the global transportation cost rate, each within the elastic as well as the non-elastic market environments are presented in Section 5. Section 6 conveys conclusions and further thoughts.

2. Background

There are three well-known pricing policies practiced by the processors in agricultural input markets: free on board pricing (FOB), uniform delivered pricing (UD), and optimal discriminatory (OD) (where the transport costs are shared equally between the processor and the farmer) pricing [20]. Until the early 1990s, the researchers' views on the choice of pricing policies by processors and their market implications fundamentally relied on non-mathematical models [21–23] but showed the more frequent practices of UD and

OD pricing relative to FOB pricing. Refs. [11,12] were the first well-known studies that analytically modeled the spatial pricing of firms. Both studies suggested that UD pricing policies are likely to be observed in industries where the transportation costs are high but also in industries with low transportation costs, while FOB is likely for intermediate market settings. Ref. [13] highlights that the presumed farmers' output supply elasticity of price in the models of [11,12] is assumed to be perfectly inelastic, leading to the processors' policies being biased in favor of UD pricing. Using a supply function with strictly positive (unitary) price elasticity for farms, Ref. [13] suggests that contrary to the model proposed by [11], the FOB pricing policies emerge as the equilibrium under market settings with low transportation costs. Mixed FOB–UD policies (one of processors chooses an FOB and the other a UD policy) are Nash equilibria in market settings with high transportation costs, and UD pricing emerges when shipping costs are high relative to the value of the finished product, for example for markets that are nearly monopsonistic in nature. Ref. [14] adopts the model proposed by [13] for specific processor objectives and shows that the coexistence of processors with different objective functions (one of processors is a profit maximizer and the other aims to simultaneously maximize its profit and the farmers' profits) are likely to give rise to mixed pricing policies. According to [14] UD (FOB), pricing is chosen in markets where transportation costs are small (large) relative to the net value of the primary product. A mixed FOB–UD pricing equilibrium emerges for an intermediate market setting.

Although these models present strong analytical basics, they fail to reflect complex environmental features like dynamic interactions as well as pricing policies with various degrees of transport cost absorptions. Agent-based models (ABMs) serve as computational laboratories with bottom-up approaches to understand market outcomes emerging from autonomously deciding and interacting agents. ABMs employ various learning methods. However, individual agents must be equipped with appropriate adaptive decision mechanisms to successfully simulate such emergent behavior at the system level [17]. In spatial competition contexts, each pricing agent needs to dynamically keep up with the changes in the prices of other agents. According to [24], there are three main approaches to learning: supervised, unsupervised, and reward-based learning. In supervised learning, an agent deals with the problem of learning through training with a series of input and output pairs. A teacher or supervisor steers the learning progress by providing feedback on the success. Deep neural networks (DNNs) [25] are typically examples of supervised learning. In unsupervised learning, no feedback is provided. The data mining methods clustering and discovery are examples of unsupervised learning. The reward-based learning methods are divided into two subsets: reinforcement learning (RL) and stochastic search methods such as genetic algorithms (GAs).

The study of market power in a broad range of studies in computational economics is often carried out using genetic algorithms [7,26–29]. Agents using a genetic algorithm require less prior competence in specific tasks [28]. Such evolutionary algorithms can be quite useful for some classes of complex problems, especially when the problem is non-trivial to deal with. Refs. [7,30] were among the first studies that applied ABMs with incorporation of *GAs* to investigate the spatial pricing policy of firms in agricultural procurement markets. They found that UD pricing is an equilibrium behavior under high and medium transport rate market regimes, and OD emerges in markets with high shipping costs. In contrast, FOB pricing does not emerge in equilibrium. Ref. [31] investigated the joint selection of location and price policy by processing firms and found that, when buyers have the flexibility to jointly choose their locations and pricing policies, farm product procurement markets are both more competitive and more efficient. In particular, they reach the result that pricing policies close to FOB pricing *re-emerge* as equilibrium strategies in market settings with low transport costs. The representative articles in the context of the firms' spatial pricing are depicted in Table 1.

Table 1. Firms' pricing policies in agricultural input markets from different agricultural economic studies' points of view.

Work by	Pricing Game	Supply Elasticity	Specific Firm Character	Equilibrium by High Transport Cost	Equilibrium by Medium Transport Cost	Equilibrium by Low Transport Cost
[11]	Repeated Game	Constant = 0	No	UD	FOB	UD
[13]	Static	Constant = 1	No	UD	FOB-UD	FOB
[14]	Static	Constant = 1	IOF or COOP	FOB	FOB-UD	UD
[30]	Repeated Game	Variable	No	OD	UD	UD
[31]	Repeated Location and Pricing Game	Variable	No	OD	UD	Close to FOB

From a critical point of view, the interpretation of the dynamics of genetic algorithms as *individual* learning processes is not in all cases clear [32]. Hence, instead of elaborating on GAs, in this paper, we opt for using a reward-based method, i.e., reinforcement learning. Thus, we made use of deep neural networks (DNNs) to develop RL pricing agents in agricultural input markets. Choosing DNNs in our study was due to the capability of DNNs to process unprecedented amounts of data [25]. Hence, we combine the application of RL with DNNs. The joint application of RL and DNNs is a solution first presented by [33] to overcome the course of dimensionality issue in market interactions with rich decision spaces. We used a well-known RL algorithm, i.e., the Q-learning algorithm [34]. Q-learning agents elaborate decisions based on the notion of dynamic programming [35] to solve optimization problems by combining solutions to sub-problems. Each agent solves each sub-problem just once and saves its answer in a memory table (named Q-table), to avoid the re-computation. The entries of tables are called Q-values. The evaluations of Q-values are then iteratively reinforced and improved as the game is played more and more. As the basic Q-learning algorithm needs huge values of storage table capacity, it is predominantly applicable to just problems with small decision spaces [36]. Ref. [33] was the first to introduce the joint usage of Q-learning together with the utilization of deep artificial neural networks (DNNs) to overcome the issue of memory storage capacity. Hereby, the deep learning model takes the role of storage tables and can predict the Q-values without usage of any tables.

To examine the results of deep reinforcement learning processors (which act in line with [33]), in our study, we utilized unsupervised agents as well. The behavior of supervised agents serves as a game theoretic benchmark to assess the performance of the unsupervised agents.

3. Market Spatial Setting and Processing Firms' Pricing Components

In this section, we describe the spatial setting of the underlying simulated market in our study and introduce processing firms' pricing components. We presume two price setting processors (as purchasers) located on a one-dimensional space. The region is discrete in space, consisting of a grid of cells with each cell occupied by exactly one farm (as supplier) and locations are accessible by X-Y coordinates, where X = {−100, . . . , −1, 0, 1, . . ., 100} and Y = 0. The location of processors in each simulation is fixed on the points (−100, 0) and (+100, 0). To normalize the factor distance, each distance between two next to each other points of the grid world is normalized by dividing by 100, so that each farm's distance to its direct neighbor is equal to 0.01 and the fixed processors' distances to each other is equal to 2.

Processors are price-setting profit maximizing processors. A general price equation is assumed to describe the net price per unit quantity of supply (local price) received by farmers at each location:

$$u_p(d_{sp}) = m_p - t\alpha_p d_{sp} \qquad (1)$$

In our study, the vector (m_p, α_p) is the representative of a processor's pricing policy describing the price for a farmer at the processor's location by term m_p and the share of transportation cost absorbed by each farm due to the spatial differences of agents expressed by the term α_p. $u_p(d_{sp})$ is the local price of processor each supplier farm receives at its location point s; d_{sp} is the distance between processor p and the supplier farm s; and t describes a global variable for transportation cost rate. We limit the maximum possible values for product price ρ_p of processors in the downstream market via normalization equal to 1. The price policy parameters of agents m_p and α_p have discrete values between 0 and 1 with predetermined increments equal to 0.01. In line with the predecessor studies, we assume that suppliers are price takers and will produce the amount q_{sp}

$$q_{sp} = u_p(d_{sp})^\varepsilon \qquad (2)$$

based on the local price they receive and the factor ε, which represents the price elasticity of supply. Note that the local prices received by each farm must be positive. However, the deep learning processors in our study are learning agents, who are free to purchase the raw product even if it does not yield a positive local profit for them. Hence the set of potential suppliers for each processor was not limited by us beforehand within the space by any marginal location.

After submitting the processors' bids to potential suppliers, each processor will earn a local profit Π_{sp} with its ultimate supplier calculated as

$$\Pi_{sp} = \rho_p - u_p(d_{sp}) - t d_{sp} \qquad (3)$$

Ultimately, each processor's profit in our model is the sum of all local profits of its contracted suppliers. If two processors submit equal local bids to a farm, the supply of that farm is shared between the contracted processors (market overlap).

4. Learning Model

In this section, we convey the description of the two types of learning agents in our study, i.e., the unsupervised agents and the supervised agents.

4.1. The Unsupervised Agents

The unsupervised agents in our study are deep Q-learning pricing agents. The theory of Markov decision processes (MDPs) offers a framework for modeling the decision-making procedure by agents in the context of Q-learning. Q-learning uses MDPs for world representation. A MDP [37] is a tuple (S, A, T, R), where S is the set of states, A is the set of actions, T is a transition function $S \times A \times S \rightarrow [0, 1]$, and R is a reward function $S \times A \rightarrow R$. The transition function defines a probability distribution over the next states as a function of the current state and the agent's action. The reward function defines the reward the agent receives when selecting an action at a given state. Solving MDPs consists of finding a policy function μ, $\mu : S \rightarrow A$, which maps states to actions. An optimal policy maximizes the sum of future rewards r, discounted by factor γ, over time t. The optimal way for agents to learn the optimal policy is learning the optimal value function [38]. A Q-function is defined as the expected discounted reward given the agent takes a certain action a *in* state s following policy μ.

$$q^\mu(s,a) = E\left(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = S, a_t = a, \mu\right) \qquad (4)$$

The optimal Q-function is defined as $q^*(s,a) = max_\mu q^\mu(s,a)$. It satisfies the Bellman [35] optimality equation:

$$q^*(s,a) = \sum_{s' \in S} T(s,a,s')\left[r(s,a,s') + \gamma max_{a'} q^*(s',a')\right] \forall s \in S \; \& \; a \in A \qquad (5)$$

Equation (5) states that the optimal value of taking *a* in *s* is the expected immediate reward from undertaking *a* plus the expected discounted maximum value attainable from the next state *s'*. Once q* values corresponded to actions in each state are available, the optimal policy will be returned in every state by reinforcing the action with the largest optimal q-value.

$$\mu^*(s) \leftarrow \arg max_a q^*(s,a) \qquad (6)$$

The optimal policy μ^* of agent in each state would be typically assigning probabilities to actions that obtain higher Q-values. A broad range of single and multi-agent RL algorithms are derived from the basic Q-learning developed by [34]. A Q-learning agent maintains the value of each possible action in every state of the environment. These are called Q-values and are stored in a table. The evaluations of the quality of particular actions at particular states are iteratively improved. The agent, subject to an *error*, selects the most favorable action (the action that gives already the maximum Q-value in his current state) *a* in its current state *s*. For example, in a so-called *epsilon-greedy* policy (which is used in our simulations) an agent chooses a random action (error) with a small probability *epsilon* and with a probability equal to *1- epsilon* decides to take the action, which gives already the maximum Q-value in his current state. This parameter is set to 0.2 in our study. Then, it perceives the consequence of this action in form of the new state of the environment *s'* and its reward *r*. Through this reward, the agent validates the significance of its last action and updates its Q-value. Hence, Q-learning turns into an iterative approximation procedure. The agent starts with an arbitrary Q-function, observes transitions $(s_k, a_k, s_{k+1}, r_{k+1})$, and after each transition, updates the Q-function according to

$$q_{k+1}(s_k, a_k) = q_k(s_k, a_k) + \alpha_k\left[r_{k+1} + \gamma max_{a'} q_k(s_{k+1}, a') - q_k(s_k, a_k)\right] \qquad (7)$$

The term within the bracket on the right-hand side of Equation (7) is the difference between the current estimate of Q-value of (s_k, a_k) and the updated estimate of (s_k, a_k). Parameter settings influence the quality of learning. For example, setting factor α_k to 0 means that the Q-values are never updated; hence, nothing is learned. Setting a high value such as 0.9 means that learning can occur quickly. This parameter is set to 1.0 in our study. The discount factor γ describes how an agent will evaluate the rewards, which are obtained afterwards. If the discount factor meets or exceeds 1, the q-values may diverge. This parameter is set to 0.5 in our study.

The classical Q-learning algorithms are predominantly applicable to small problems only, e.g., games with two players and two possible actions per each player. By increasing the interaction, the domain tabular storage of Q-functions for agents becomes economically infeasible, i.e., impractical. The number of actions and states in a real-life environment can be thousands upon thousands, making it extremely inefficient to manage Q-values in a table. Recent advances in deep neural networks (DNNs), especially in deep learning, have enabled the application of Q-learning algorithms to large-scale decision problems [33,39]. In this case, one can use DNNs to predict Q-values for actions in a given state instead of using tables. As an alternative for initializing and updating a Q-tables in Q-learning processes, we initialized and trained a neural network model to predict Q-values. DNNs consist of artificial neurons that receive and process input data. Data are passed through the input layer, the hidden layer, and the output layer to predict complex patterns [40,41]. The layers of the neural network used in our study comprise the following. The input layer consists of four nodes comprising a tuple of m_p and α_p of both players, which represents the state of the world in our study. There are three hidden layers in the neural network architecture of our study, each of them consisting of 50 neurons, respectively. The output layer consists of five nodes comprising the number of actions that each processor can undertake by

observing the state of the world. Each of the five actions enable processors to decide upon the change in each element of their pricing vectors (m_p, α_p) with the preferred increment sizes $\delta = 0.01$, in line with the following gradients: $[(-\delta, 0), (0, -\delta), (0, 0), (0, \delta), (\delta, 0)]$.

The DNN minimizes the error function (called loss function) presented in Equation (8) through the course of learning, i.e., the square value of the difference between the predicted $q_k(s_k, a)$ and the target q values ($r_{k+1} + \gamma max_{a'}q_k(s_{k+1}, a')$):

$$loss_{k+1} = \left\{ r_{k+1} + \gamma max_{a'}q_k(s_{k+1}, a') - q_k(s_k, a) \right\}^2 \tag{8}$$

The unsupervised agents' (deep Q-learning pricing agents) learning algorithm is shown in the following in pseudo-code format:

I.   Initialize a DNN model;
II. Initialize a list for memorizing (state of the world, action, new state of the world, reward) in each step of the game;
III. In each step of the game:

   a.   Observe the state of the world comprising all processor firms' prices;
   b.   Demand the DNN model to predict the Q-value of each action from the state of the world;
   c.   If you are not in error mode:

   i.   Choose the action with the highest Q-value;

   d.   Otherwise:

   i.   Choose a random action;

   e.   Adjust the pricing policy based on the chosen action;
   f.   Participate in the spatial competition by applying the determined pricing policy;
   g.   Each farmer decides whether to connect and deliver to which processor based on the processors' determined pricing policies;
   h.   Collect the input product from the connected farmers based on the pricing policy;
   i.   Pay the transportation cost according to distance to each farmer;
   j.   Process the input product and sell the processed product in the downstream market;
   k.   Calculate the final pay-off;
   l.   Set the final pay-off as reward;
   m.   Observe the new state of the world comprising all processor firms' prices;
   n.   Extend memory based on new information: (state of the world, action, new state of the world, reward);
   o.   For states of the worlds in the memory list:

   i.   Demand the DNN model predict the Q-value of each action from the new state of the world;
   ii. Set the highest Q-value among actions as the *Max_New_state_Q_Value*;
   iii. Compute the Q-value of the chosen action from the state of the world according to equation: *reward + discount_factor * Max_New_state_Q_Value*;

   p.   Train the DNN model (1 epoch) by using the states of the world as input and computed Q-values of each action from the state of the world as output.

Note that the unsupervised agents decide simultaneously upon updating their pricing policies in each step of the game. The training code to replicate the training process is included in the Supplementary Materials of this paper.

4.2. The Supervised Agents

In order to examine the performance of the unsupervised agents introduced in the previous section, we use a game theory approach analogous to the sequential move adjustment process in [19,42]. We presume that when processors decide to become involved in spatial competition, they start pricing from an arbitrary price point but proceed to sequentially apply the best response priced to each other. The processors' mutual best

responding is expected to lead us to the same termination points as the Nash equilibrium of the game [19]. Here, we initiate a course of best response sequential play between agents from the point ($m_p = 0.50$, $\alpha_p = 0.50$) and let the processors undertake moves subject to the assumption that both agents know to choose the most profitable pricing policy from all $10 \times 10^2 \times 10 \times 10^2$ possible combinations of their (m_p, α_p), given the pricing policy of the opponent. A complete list of best response policies given the price of the opponent in the market environment presented in Section 3 was computationally prepared in the tables and was fed into the agents by us as the supervisors before the sequential move game began. The supervised agents' learning algorithm is shown in the following in pseudo-code format:

I.   In each step of the game:

    a.   Observe the state of the world comprising all processor firms' prices;
    b.   Select the best response pricing policy given the opponent's prices based on the information provided by supervisor;
    c.   If the same state of the world is twice observed:

        i.   Report the sequence of repeated states of the world comprising all processor firms' prices as equilibria.

Note that supervised agents do not decide simultaneously. Each of the supervised agents decides upon its pricing policy at each step of the game when it is its turn. The training code used to replicate the training process is included in the Supplementary Materials of this paper.

Furthermore, we do not presume that the supervised agents necessarily report the unique state of the world as the Nash equilibrium in the market. In spatial competition with the incorporation of transport cost absorption policies, there is no guarantee for an existing unique verifiable price combination of processors as the mutually best response of players to each other. In such circumstances, the cyclic price behaviors can take the place of Nash equilibria. This phenomenon happens due to the discontinuous nature of best response functions of players in agricultural markets with transport cost absorptions and is discussed in the literature [43–47].

5. Simulation Results

The dependent variable of our study is set to be each of two processors' pricing vectors (m_p, α_p) in competition with the opponent in the market environment. The explanatory variable for the pricing policies of the pricing agents in the existing literature of spatial competition is a ratio called the *importance-of-space* measured by $s = t \times D/\rho$, i.e., transport cost rate (t) multiplied by distance to the competitor (D) divided by the net value of product being sold at the downstream market (ρ) [5]. The more the ratio s increases, the more the competition between processors diminishes, to the point where they are eventually spatially isolated monopsonies. The more the ratio s decreases, the more intensive the competition between processors becomes. In order to alter the value of s as an explanatory variable in our study, we exogenously changed the parameter t in the range of the values within the list t = [0.01, 0.2, 0.4, 0.6, 0.8, 1.0]. In addition, we conducted the simulations, once with the factor price elasticity of supply ε equal to 1 (which represents a strictly positive (unitary) price elasticity for farms) and once with the factor price elasticity of supply ε equal to 0.01 (which represents an extremely non-elastic market supply). The near to zero ε parameter could reflect the fact that, in reality, farmers might have limited flexibility to substitute outputs, creating a relatively inelastic supply in the short-term [48]. Each simulation was conducted through 10,000 steps and repeated by changing the parameters, agent types (supervised or unsupervised), and t and ε, according to the abovementioned values. Figures 1 and 2 show the finalized m_p, α_p outcome of the simulations by unsupervised agents at the final 1000 steps of the agents' interactions, as well as the equilibrium values of m_p, α_p obtained by supervised agents by setting the price elasticity of supply ε equal to 1 and 0.01, respectively for the selected values of transport costs.
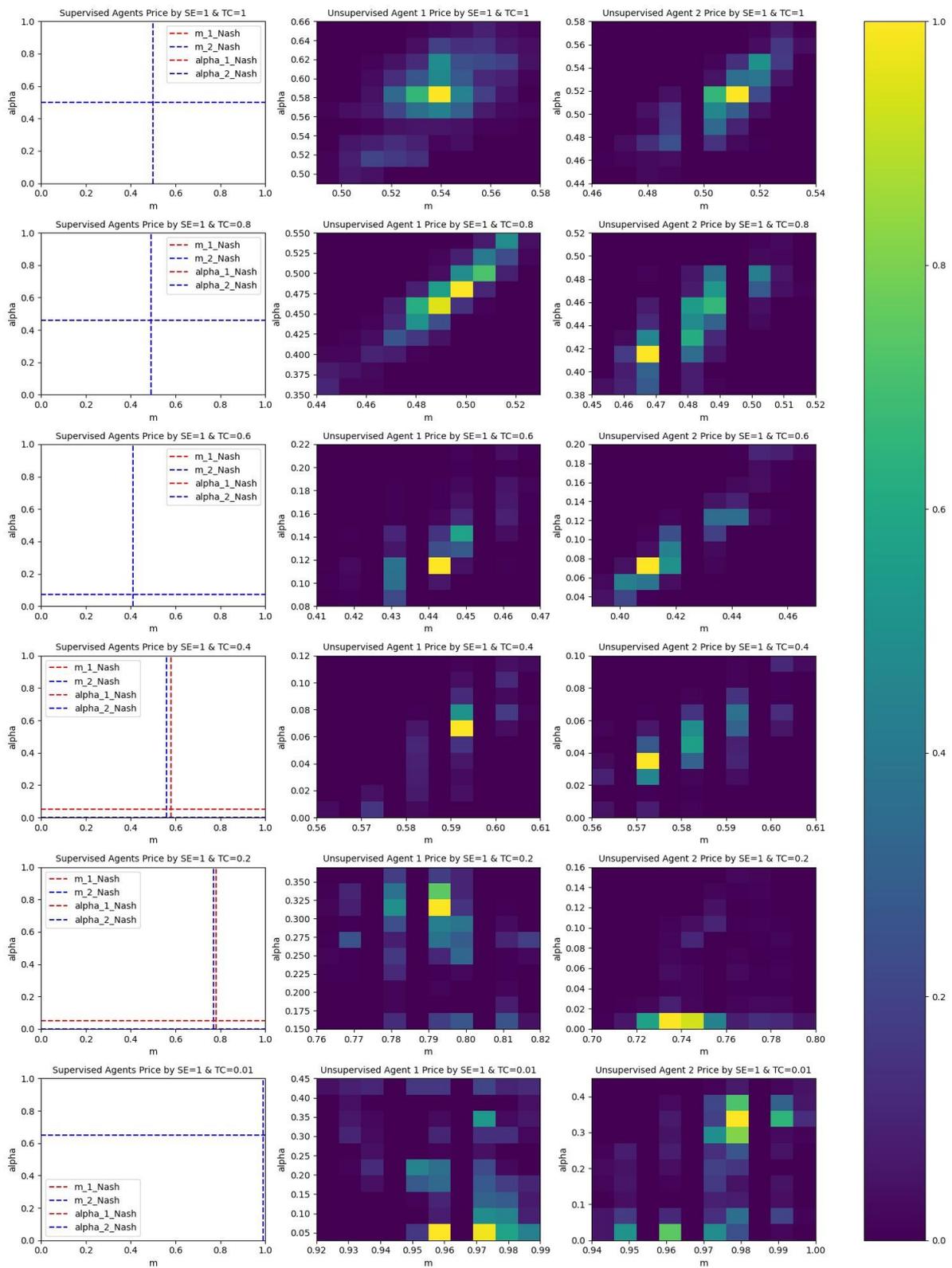
Figure 1. Pricing of the duopsony for selected values of *t* in the case of an elastic market ($\varepsilon = 1$).
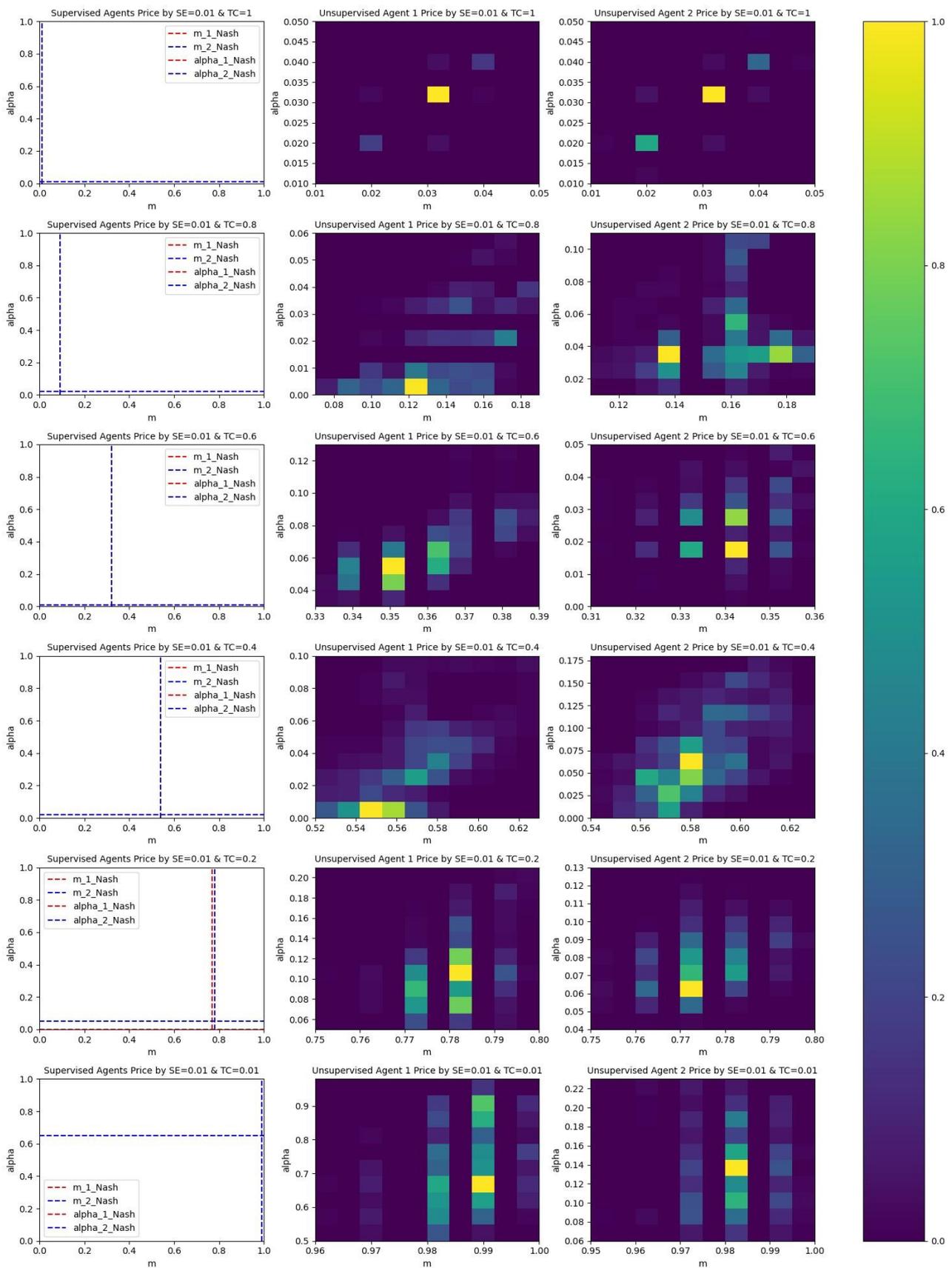
Figure 2. Pricing of the duopsony for selected values of *t* in the case of a non-elastic market ($\varepsilon = 0.01$).

The left-side panels of Figures 1 and 2 reveal that the spatial market system does comprise a verifiable unique Nash equilibrium with practically symmetric prices for both processor firms throughout all the market settings. This means that the case of cyclic price wars, i.e., the non-existence of the Nash equilibrium is not observed in our study through the underlying market environment and the ranges of selected parameters for transport rate and the price elasticity of supply.

The middle and the right-side panels of Figures 1 and 2 demonstrate the frequency of m_p, α_p variable combinations obtained by unsupervised competitor agents. The lighter a point within the m-α plane, the more frequent the variable combination is observed as the outcome of the game by unsupervised agents. The points outside of the depicted minimum and maximum of the m-α plane are non-observed pricing policies in the final steps of the agents' interaction.

The findings, with regard to the obtained equilibrium prices in Figures 1 and 2, are twofold: first, in all the market settings within the selected parameters for transport rate and the price elasticity of supply, the unsupervised deep learning agents approximated the Nash equilibrium target m_p, α_p points in line with the supervised agents' outcomes, in the course of market interactions, by learning. This approximation is accomplished with a minor exception when the parameter t is set extremely small ($t = 0.01$) both in elastic ($\varepsilon = 1$) and non-elastic ($\varepsilon = 0.01$) markets. The simulations show that in the extreme cases of setting the lowest possible transportation rate, unsupervised agents proceed on the course of overbidding the mill prices m_p until they achieve the highest levels of mill prices, i.e., $m_p \approx 1.0$, which is completely in line with the Nash equilibrium outcome of the game by supervised agents; however, whereas the supervised processors, both in the case of elastic market as well as in the inelastic market incorporated $\alpha_p \approx 0.6$ (with regard to the absorption of transportation costs), the most frequently observed m_p and α_p values by unsupervised agents varied around $\alpha_p = 0.0$–0.4 in the elastic market and around $\alpha_p = 0.0$–0.2 by one of the pricing agents in the non-elastic market (whereas the other unsupervised agent in Figure 2 approximated the target $\alpha_p \approx 0.6$ level similar to the supervised agents' Nash equilibria points). We can interpret this outcome in relation to the little weight the deep learning might have put on the importance of the α_p (as the coefficient of transport cost) in a market where the role of transport ($t = 0.01$) is not significant. In this case, the equilibrium of the market with extremely small transportation costs is expected to recap the *Bertrand* solution with maximum mill prices $m_p \approx 1.0$ set by both processor agents and a little incentive to regulate the α_p together with roughly zero profits obtained by both processors. Despite this deliberation, it would not be surprising if the deep learning pricing agents are able to exactly settle on the target Nash equilibrium points through further learning in extra simulation steps.

The second finding with regard to the obtained equilibrium prices in our study is related to the question of which prices emerge as Nash equilibrium in spatial agricultural procurement markets?

From Figures 1 and 2, it is evident that for the cases when the transportation cost rate t is set as equal to 1.0 and 0.8, both in the elastic market and the non-elastic market, the market prices move towards the monopsonistic optimal discriminatory prices. A monopsonistic optimal discriminatory price (OD) policy comprises mill price m_p and the transport cost absorption by the monopsony processor α_p equal to the tuple $(m_p, \alpha_p) = (1/(1 + \varepsilon), 1/(\varepsilon + 1))$ [49]. Thus, where $\varepsilon = 1.0$, we expect the m and *alpha* variables to converge around the point $(m_p, \alpha_p) = (0.5, 0.5)$, and where $\varepsilon = 0.01$, we expect the m and *alpha* variables to converge around the point $(m_p, \alpha_p) = (0.01, 0.01)$. By decreasing the transportation cost rate t to 0.6, 0.4 and 0.2, we can clearly observe that the processors' pricing policies, both in the elastic market and the non-elastic market, converge towards setting the farm gate price and bearing all the transportation costs, i.e., $\alpha_p \approx 0.0$. Only in the extremely competitive market setting where the spatial feature of the market almost does not matter, i.e., $t = 0.01$, do the processors' prices tend to involve the high absorption of (the actually small) transport costs to be carried by the farms, i.e., close to FOB policies.

These outcomes are, on the one hand, thorough with regard to the individual learning and the game theory foundation of the firms' policies. On the other hand, our simulation outcomes support the prevalence of pricing policies with the absorption of high portions or the entire transport charges through the processing firms in a broad range of market settings, while proposing the emergence of policies close to FOB both in extremely competitive elastic and inelastic market settings.

6. Conclusions and Further Discussion

After comparing the two major simulation results in Section 5, we reflect on how the research questions of our paper have been answered, on what main lessons can be drawn, and what further research needs evolve from our study. Price formation in agricultural procurement markets is a complex dynamic process with multiple agents and interactions. Agent-based simulation models are thought to simulate the actions and interaction of autonomous agents in complex environments. However, individual agents must be equipped with appropriate adaptive learning mechanisms to successfully simulate the emergent pricing policies at the system level. In this paper, we introduced a deep learning dynamic model of pricing to the context of agricultural markets in large-scale strategy spaces. We designed experiment runs (by changing the transport cost rate and price elasticity of supply) to examine whether the deep learner agents can converge to optimal pricing policies (to answer RQ 1) in different spatial market settings. To provide decent criteria to answer the RQ 1, we designed markets comprising cognitively equipped supervised agents, which are able to pinpoint the market Nash equilibria (to answer RQ 2). Our simulation results showed that unsupervised deep learning firms are capable of converging towards Nash equilibria prices. The significance of the deep learning agents is not only the convergence to the targeted theoretical points in line with the supervised agents pricing policies, but it is more important that the deep learning agents can inherently keep up with changes in the environment through considering the state of the environment (consisting of the pricing policy vectors of all market processors) in their input. Deep learning agents constantly move towards the optimal action by generating the right decision in their model output. This means that if, e.g., one of the competitor processors' price vector changes and settles on a stationary pricing vector, then a deep learning pricing agent will also update its policy to converge on a policy that is the best response to the opponent processor's policy. Based on the incorporated results obtained both from supervised and unsupervised agents (which correlate significantly with each other), our research provides insights into equilibrium pricing policies in agricultural input markets (to answer RQ 3). Our simulations support the theoretical and empirical reflections of the recent existing studies (introduced in the background section) with regard to the emergence of optimal discriminatory and uniform delivery pricing policies in a broad ranges of agricultural input markets with a high and intermediary amount of importance being placed on space. Moreover, our results show the emergence of free on board pricing policies both in elastic and inelastic agricultural input markets with a small amount of importance being placed on space.

While our study's objective, with regard to the defined RQs within this paper, is attained, there are limits to the scope of our research which need to be addressed in future studies. Our simulations are still limited to two agents and fixed firm locations. Real-world markets can involve pricing with the presence of multiple agents and with incorporating agents who decide upon the joint selection of pricing and location. Other real-world constraints could also be imposed on food processors and farmers, including limitations on production capacity, legal constraints through price rules, objectives of specific actors rather than profit maximization (Section 2), etc. Increasing the complexity of the market environment requires more enhanced versions of deep learning models than the one presented in this paper. The current deep Q-learning model still necessitates agents to explore a vast number of world states capable of achieving the convergence criteria in line with the proposed Nash equilibria. Utilizing hierarchical deep learning agents in agricultural markets could be a future step of research for coping with the computational

burdens triggered through additional market features. Rational agents in reality guide their decisions using hierarchical elaboration rather than undertaking exhaustive searches of the whole decision space. Elaborating on hierarchical learning in such complex systems can prevent agents from changing their policies in an arbitrary fashion and can reduce the dimensionality of interactions.

References

1. Sexton, R.J. Imperfect Competition in Agricultural Markets and the Role of Cooperatives. *Am. J. Agric. Econ.* **1990**, *72*, 709–720. [CrossRef]
2. Sexton, R.J. Market Power, Misconceptions, and Modern Agricultural Markets. *Am. J. Agric. Econ.* **2012**, *95*, 209–219. [CrossRef]
3. Rogers, R.T.; Sexton, R.J. Assessing the Importance of Oligopsony Power in Agricultural Markets. *Am. J. Agric. Econ.* **1994**, *76*, 1143–1150. [CrossRef]
4. Durham, C.A.; Sexton, R.J.; Song, J.H. Spatial Competition, Uniform Pricing, and Transportation Efficiency in the California Processing Tomato Industry. *Am. J. Agric. Econ.* **1996**, *78*, 115–125. [CrossRef]
5. Alvarez, A.M.; Fidalgo, E.; Sexton, R.; Zhang, M. Oligopsony Power with Uniform Spatial Pricing: Theory and Application to Milk Processing in Spain. *Eur. Rev. Agric. Econ.* **2000**, *27*, 347–364. [CrossRef]
6. Huck, P.; Salhofer, K.; Tribl, C. Spatial Competition of Milk Processing Cooperatives in Northern Germany. In Proceedings of the International Association of Agricultural Economists Conference, Queensland, Australia, 12–18 August 2006.
7. Graubner, M.; Koller, I.; Salhofer, K.; Balmann, A. Cooperative versus Non-Cooperative Spatial Competition for Milk. *Eur. Rev. Agric. Econ.* **2011**, *38*, 99–118. [CrossRef]
8. Hamilton, S.F.; Sunding, D.L. Joint Oligopsony-Oligopoly Power in Food Processing Industries: Application to the US Broiler Industry. *Am. J. Agric. Econ.* **2020**, *103*, 1398–1413. [CrossRef]
9. Deconinck, K. *Concentration and Market Power in the Food Chain*; OECD Food, Agriculture and Fisheries Papers No. 151; OECD Publishing: Paris, UK, 2021. [CrossRef]
10. Jung, J.; Sesmero, J.; Siebert, R. A Structural Estimation of Spatial Differentiation and Market Power in Input Procurement. *Am. J. Agric. Econ.* **2022**, *104*, 613–644. [CrossRef]
11. Espinosa, M.P. Delivered pricing, FOB pricing, and collusion in spatial markets. *RAND J. Econ.* **1992**, *23*, 64–85. [CrossRef]
12. Kats, A.; Thisse, J.F. Spatial oligopolies with uniform delivered pricing. In *Does Economic Space Matter?* Ohta, H., Thisse, J.-F., Eds.; St Martins Press: New York, NY, USA, 1993; pp. 274–296. [CrossRef]
13. Zhang, M.; Sexton, R.J. FOB or Uniform Delivered Prices: Strategic Choice and Welfare Effects. *J. Ind. Econ.* **2001**, *49*, 197–221. [CrossRef]
14. Fousekis, P. Free-on-board and Uniform Delivery Pricing Policies in a Mixed Duopsony. *Eur. Rev. Agric. Econ.* **2011**, *38*, 119–139. [CrossRef]
15. Tesfatsion, L. Chapter 16 Agent-Based Computational Economics: A Constructive Approach to Economic Theory. In *Handbook of Computational Economics*; Tesfatsion, L., Judd, K.L., Eds.; Elsevier: Amsterdam, The Netherlands, 2006; Volume 2, pp. 831–880. ISSN 1574-0021, ISBN 9780444512536. [CrossRef]
16. Grimm, V.; Railsback, S.F. *Individual-Based Modeling and Ecology*; Princeton University Press: Princeton, NJ, USA, 2005.
17. Kirman, A. Learning in ABMs. *East. Econ. J.* **2011**, *37*, 20–27. [CrossRef]
18. Weiss, G. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*; MIT Press: Cambridge, MA, USA, 2000.
19. Fudenberg, D.; Levine, D.K. *The Theory of Learning in Games*; MIT Press: Cambridge, MA, USA, 1998.
20. Beckmann, M.J. Spatial Price Policies Revisited. *Bell J. Econ.* **1976**, *7*, 619–630. [CrossRef]

21. Scherer, F.M. *Industrial Market Structure and Economic Performance*, 2nd ed.; Rand-McNally College Publishing Co.: Chicago, IL, USA, 1980.
22. Greenhut, M.L.; Ohta, H. Monopoly Output under Alternative Spatial Pricing Techniques. *Am. Econ. Rev.* **1972**, *62*, 705–713.
23. Greenhut, M.L. Spatial pricing in the USA, West Germany and Japan. *Economica* **1981**, *48*, 79–86. [CrossRef]
24. Panait, L.; Luke, S. Cooperative Multi-Agent Learning: The State of the Art. *Auton. Agents Multi-Agent Syst.* **2005**, *11*, 387–434. [CrossRef]
25. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006.
26. Vallée, T.; Başar, T. Off-Line Computation of Stackelberg Solutions with the Genetic Algorithm. *Comput. Econ.* **1999**, *13*, 201–209. [CrossRef]
27. Alemdar, N.M.; Sirakaya, S. On-line Computation of Stackelberg Equilibria with Synchronous Parallel Genetic Algorithms. *J. Econ. Dyn. Control.* **2003**, *27*, 1503–1515. [CrossRef]
28. Arifovic, J. Genetic algorithm learning and the cobweb model. *J. Econ. Dyn. Control.* **1994**, *18*, 3–28. [CrossRef]
29. Vriend, N.J. An illustration of the essential difference between individual and social learning, and its consequences for computational analyses. *J. Econ. Dyn. Control* **2000**, *24*, 1–19. [CrossRef]
30. Graubner, M.; Balmann, A.; Sexton, R.J. Spatial Price Discrimination in Agricultural Product Procurement Markets: A Computational Economics Approach. *Am. J. Agric. Econ.* **2011**, *93*, 949–967. [CrossRef]
31. Graubner, M.; Sexton, R.J. More competitive than you think? Pricing and location of processing firms in agricultural markets. *Am. J. Agric. Econ.* **2022**, *105*, 784–808. [CrossRef]
32. Brenner, T. *Agent Learning Representation-Advice in Modelling Economic Learning*; Max Planck Institute for Research into Economic Systems: Jena, Germany, 2005. [CrossRef]
33. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef]
34. Watkins, C.J. Learning from Delayed Rewards. Ph.D. Thesis, Cambridge University, Cambridge, UK, 1989.
35. Bellman, R. *Dynamic Programming*; Princton University Press: Princton, NJ, USA, 1957.
36. Busoniu, L.; Babuska, R.; De Schutter, B. Multi-agent reinforcement learning: An overview. In *Innovations in Multi-Agent Systems and Applications—1, Studies in Computational Intelligence*; Srinivasan, D., Jain, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; Volume 310, pp. 183–221. [CrossRef]
37. Howard, R. *Dynamic Programming and Markov Process*; MIT Press: Cambridge, MA, USA, 1960.
38. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 1998.
39. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484–489. [CrossRef]
40. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
41. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef] [PubMed]
42. Maskin, E.; Tirole, J. Markov perfect equilibrium. *J. Econ. Theory* **2001**, *20*, 191–215. [CrossRef]
43. Beckmann, M.J. Spatial Oligopoly as a Noncooperative Game. *Int. J. Game Theory* **1973**, *2*, 263–268. [CrossRef]
44. Shubik, M.; Levitan, R. *Market Structure and Behavior*; Harvard University Press: Cambridge, MA, USA, 1980.
45. Schuler, R.E.; Hobbs, B.F. Spatial Price Duopoly under Uniform Delivered Pricing. *J. Ind. Econ.* **1982**, *31*, 175–187. [CrossRef]
46. Dasgupta, P.; Maskin, E. The Existence of Equilibrium in Discontinuous Games. Applications. *Rev. Econ. Stud.* **1986**, *53*, 27–41. [CrossRef]
47. Tesauro, G.; Kephart, J.O. Pricing in Agent Economies Using Multiagent Q-learning. *Auton. Agents Multi-Agent Syst.* **2002**, *5*, 289–304. [CrossRef]
48. Gardner, B. Changning Economic Perspectives on the farm problem. *J. Econ. Lit.* **1992**, *30*, 62–101.
49. Löfgren, K.G. The Spatial Monopsony: A Theoretical Analysis*. *J. Reg. Sci.* **1986**, *26*, 707–730. [CrossRef]