Quality uncertainty and market failures: an interactive model to conduct classroom experiments

María Pereda[1], David J. Poza[2], José I. Santos[1], José M. Galán[1]

INTRODUCTION

The problems that quality uncertainty and asymmetric information cause in markets are well known in Economics. Traditionally, quality uncertainty phenomena have been studied assuming that one party, i.e buyers or sellers, has better knowledge about the value of a good and exploits this information asymmetry to its own advantage. Using the automobiles market, Akerlof (1970) showed that market incentives can trigger adverse selection processes that can ruin the market.

However it is not necessary to assume asymmetric information: quality uncertainty without asymmetric information can lead to market failure, as shown by Izquierdo and Izquierdo (2007). In their model, sellers trade a commodity with a fixed quality distribution, which is a priori unknown to buyers. Buyers estimate the expected quality based on past purchases. Crucially, buyers with sufficiently many bad experiences stop buying the good (and do not update their bad expectations anymore). This leads to biased quality estimations, which can make the market collapse.

We have developed an interactive version of this model using Netlogo and HubNet technology (Wilensky,1999), specially designed to teach the effects of quality uncertainty in markets. The instructor can conduct experiments in a virtual classroom, in which students using a computer are embedded in an online market playing the role of buyers. An experiment consists of a series of trade sessions; in each session, students must decide their bidding price for the good without information about its quality distribution. The market is cleared using a fixed customizable supply function, and the students who finally purchase the good find out the actual quality of the unit they have acquired. Moreover, the instructor can explore the effect of social networks to improve market efficiency, letting students share information with their neighbors in the network.

The agent-based model of the quality uncertainty effects on markets by Izquierdo and Izquierdo (2007) was developed in Netlogo (Wilensky,1999); our version of this mode has been implemented in Netlogo too. Netlogo is an application specially designed for developing and simulating agent-based models. The readable but powerful language that allows to express complex agents' behaviors in a few code lines, the interface that facilitates build sophisticated

[1] INSISOC, Universidad de Burgos, Edif."La milanera". C/ Villadiego, s/n., 09001 Burgos, Spain
[2] INSISOC, Universidad de Valladolid, E.T.S. Ingenieros Industriales. Pº del cauce, s/n., 47011 Valladolid, Spain

models quickly and the ease of use for researches, instructors and students, have become Netlogo an application widely used within the social simulation community. We have taken advantage of the HubNet technology that is integrated in Netlogo. HubNet lets Netlogo models run in an interactive way with people, who can individually control each of the agents of the model by a networked computer. This is particularly interesting when, as in our case, you want to conduct economic experiments in classroom.

This manual explains all necessary steps in order to set up a HubNet activity of the model of quality uncertainty and run a complete trade session. First, we describe the installation and the configuration of the Netlogo model in the instructor and students' computers. Then we explain how the instructor can test the application and simulate a classroom experiment locally in her own computer. Finally we describe the instructor and students' interfaces. This manual and the model can be downloaded in the website: https://www.openabm.org/model/4214/version/1/view.

Table of Contents

INSTALLATION AND CONFIGURATION GUIDE

In order to perform the experiment, a computer room with internet connection is required. The first step is to **install Netlogo in the computers that will be used in the activity**. Netlogo can be freely downloaded from its official site: http://ccl.northwestern.edu/netlogo/. The system requirements are not demanding and Netlogo can be run in computers with different operating systems (see http://ccl.northwestern.edu/netlogo/requirements.html for additional technical information). The installation process will be the same in the instructor's computer and the students' computers: just execute the installation file and follow the instructions of the installer.

Once installed, we will need to set up Netlogo for the experiment. There will be two roles: the instructor's computer will work as the server and the students' computers will work as the clients. As of May 2014, the maximum number of clients supported by the server is 25. Notice that **there is no need to distribute the file *Quality_variability.nlogo* among the students, as it will only be run in the server** (i.e. the instructor's computer).

Setting up the instructor's computer (server)

Run the file *Quality_variability.nlogo* or, alternatively, run Netlogo and then load the file *Quality_variability.nlogo*. The server interface will be displayed. This view allows the instructor to define the parameters of the experiment and to monitor the students' actions (Figure 1).

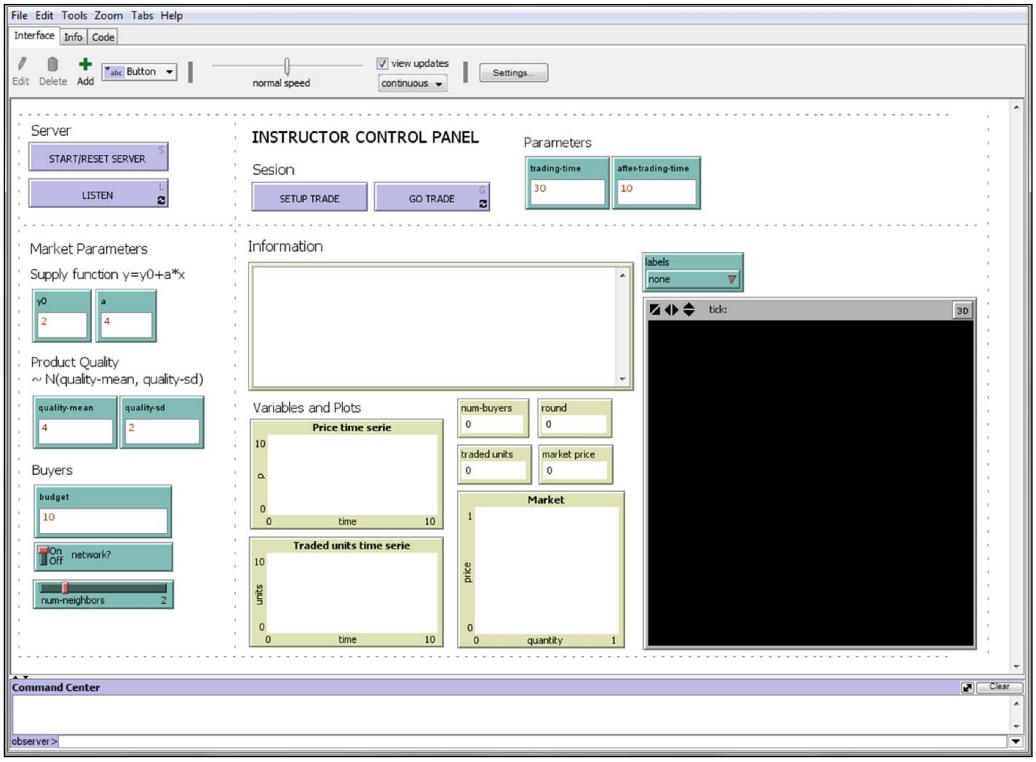


Figure 1. Instructor's interface

The first step of a classroom experiment is to start the server in the instructor's computer by clicking on the button START/RESET SERVER. The window 'Start HubNet Activity' will be displayed (Figure 2). Type a name for the session (e.g. Experiment 1). Make sure that the box 'Broadcast server location' is checked, as this will make it easier for students (clients) to connect to the server.

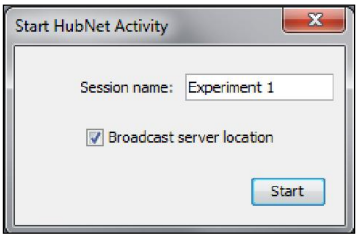


Figure 2. Start HubNet Activity

Afterwards, press the 'Start' button. The 'HubNet Control Center' window will pop up (Figure 3). This window shows the server IP (i.e. the instructor's computer's IP) and the port number that will be used. Notice that if 'Broadcast server location' was not checked in the window 'Start HubNet Activity' (Figure 2), you will need to provide the students with this information so that they can connect to the server.

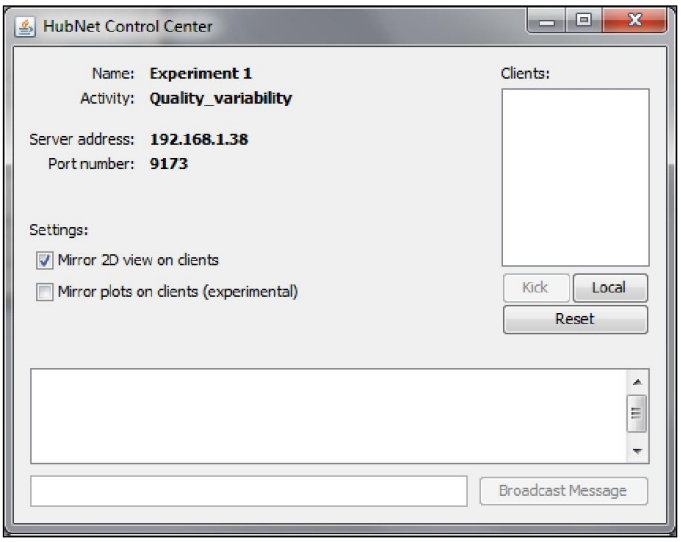


Figure 3. HubNet Control Center

Go back to the instructor's main window (Figure 1) and click on the LISTEN button. The server is now ready to receive connections from the clients. At this point, the students should be told to run HubNet in their computers to connect to the server (see instructions in the next section). As the students join the experiment, you will see their names in the Clients box (Figure 4). This information will also be displayed at the instructor's information panel (see Figure 5).

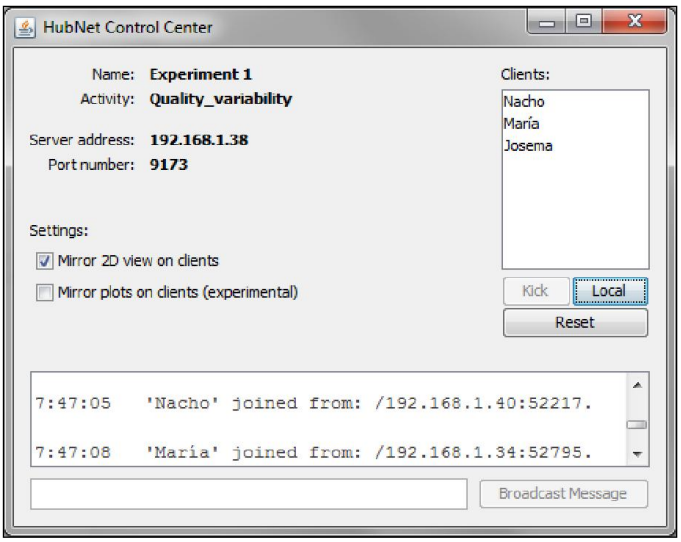


Figure 4. HubNet Control Center (once some students have connected to the server)

Once <u>all students</u> have connected to the server, the instructor must click on the SETUP TRADE button. This will create a virtual market according to the parameters chosen by the instructor (the section EXPLORING THE INSTRUCTOR'S INTERFACE provides a description of these parameters). At this point, the instructor's interface will display a graph showing as many agents as students have connected to the server (see Figure 5).

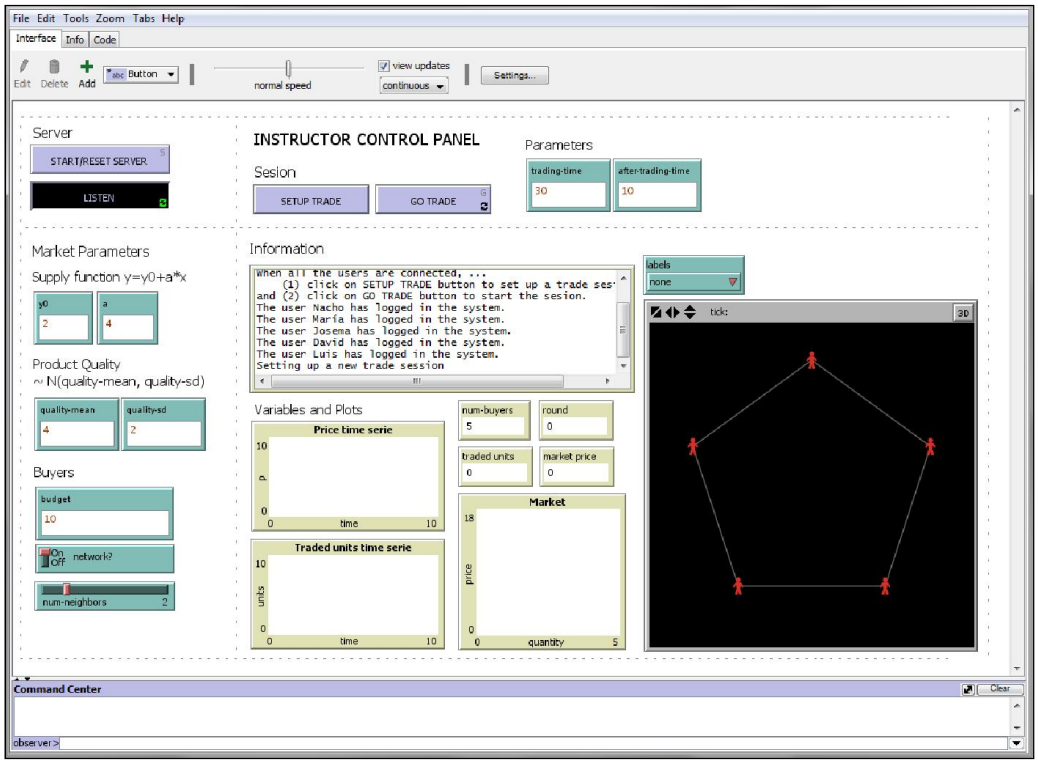


Figure 5. Information displayed in the instructor's interface

When the instructor is ready to start the activity, she may click on the GO TRADE button. The first round will start.

The experiment consists of several rounds, and each round is divided into two phases (i.e. trading and after-trading). During the trading phase, the students have the possibility of bidding. Just after the first phase have finished, the second phase starts automatically. In the after-trading phase, the system calculates the market clearing price, the item's quality of all purchases and each student's resulting benefit. When the after-trading phase is over, a new round starts.

When the instructor considers it is appropriate to finish the activity, she may click on the GO TRADE button again to stop the experiment.

Setting up the students' computers (clients)

This section together with the EXPLORING THE STUDENT'S INTERFACE section can be given out to the students as a part of the guide for preparing the classroom experiment.

After installing Netlogo from the official site http://ccl.northwestern.edu/netlogo/, proceed to locate the file "HubNet 5.0.5.exe" and execute it; the executable file should be in the folder where Netlogo was installed. Notice that the final numbers of the application depend on the version that was downloaded.

When HubNet is running under Windows, you may receive a warning from Windows Firewall. If this happens, click on the button 'Allow Access', and then the HubNet window will be displayed (Figure 6).

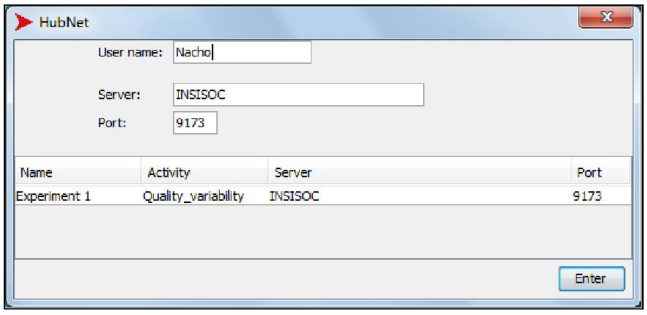


Figure 6. Executing HubNet in a student's computer. Connecting to the server

The students are to type their names in the field 'User name'. If the instructor checked the box 'Broadcast Server Location' when starting the server (Figure 2), the students will be able to see the name of the server, the port and the name of the HubNet activity (see Figure 6). Otherwise, they will have to input this information manually.

The final step is to click on the Enter button. The student's interface will appear (Figure 7). The student is now connected to the server and the game will start as soon as the instructor decides to start the trade; any message about this one or other event will be shown in the 'Information' monitor..

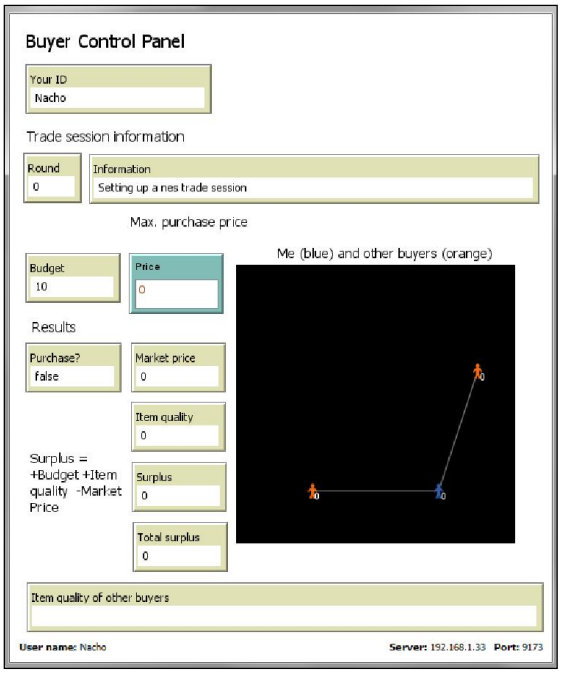


Figure 7. Student's interface

TESTING THE APPLICATION. LOCAL EXECUTION.

In order to test the application, the instructor can perform a simulation of a classroom experiment in her own computer (with no students). To do this, click on the START/RESET SERVER button and then press LISTEN. When the window 'HubNet Control Center' appears, click on the 'Local' button to create virtual students (e.g. click four times to create four students, Figure 8). Netlogo will display as many student windows as virtual students have been created.

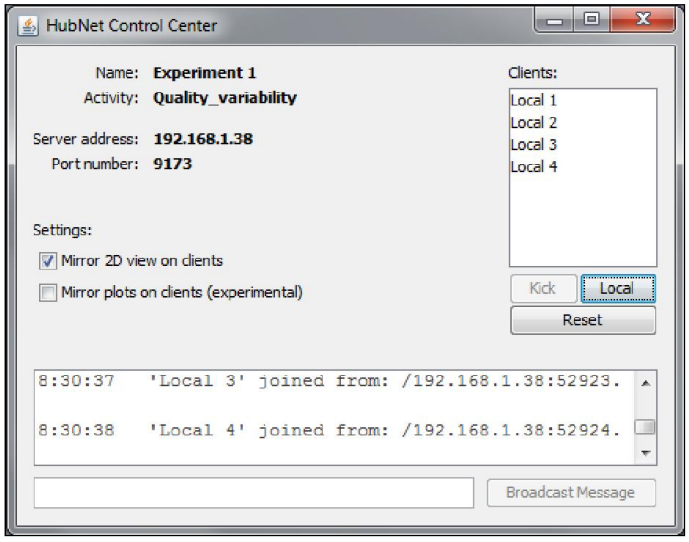


Figure 8. Creating 'virtual students' in the instructor's computer

When the instructor has finished of creating virtual students, she may return to the instructor's window, click on the SETUP TRADE button and then press GO TRADE to start a trade session.

EXPLORING THE INSTRUCTOR'S INTERFACE

The objective of the instructor's interface (Figure 9) is twofold. First, it allows defining the parameters of the experiment (e.g. supply function, timing, parameters related to the uncertainty of the quality, etc.). Secondly, it allows the instructor to monitor the evolution of the experiment (number of elapsed rounds, number of buyers, market clearing price in the current round, etc.).

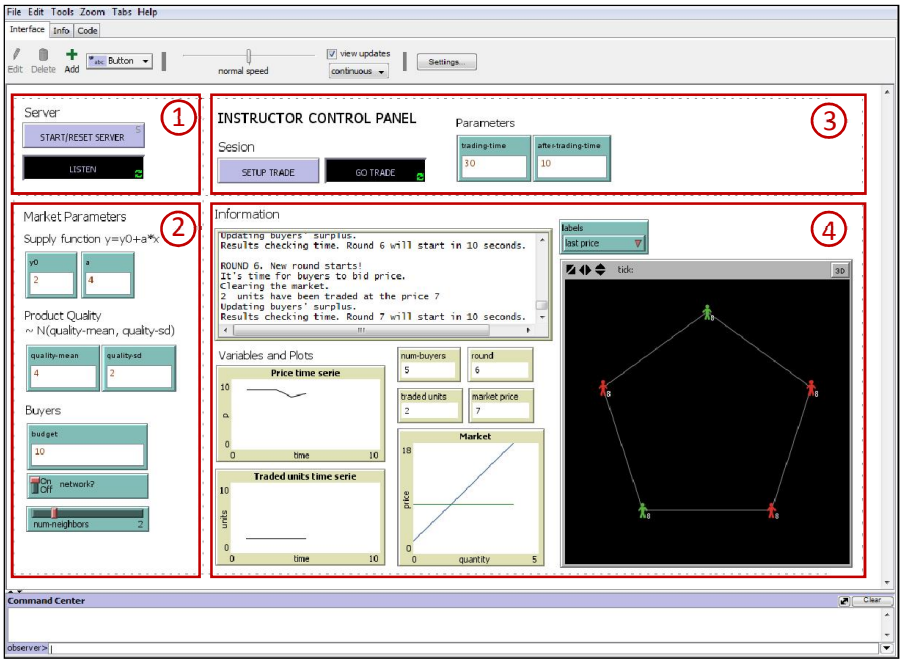
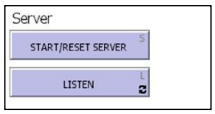


Figure 9. Exploring the instructor's interface

Server buttons (1)



These buttons allow the instructor to start and to finish the experiment. In order to initiate the HubNet server, click on the START/RESET SERVER button. The window 'Start HubNet Activity' will appear (Figure 2). Follow the instructions described in the section Setting up the **instructor's computer** (server) and then click on the LISTEN button. This will make the server listen for possible clients that are trying to connect.

Market parameters (2)

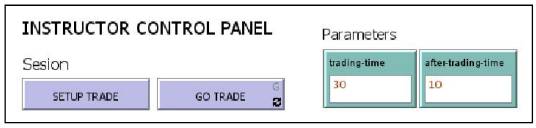
Whereas the demand side of the market depends of the students' decisions (number and value of bids), the supply side of the market is controlled by the instructor.

1. The implemented supply function is linear with parameters: intercept 'y_0' and slope '*a*'.
2. The quality of the items sold in the market is uncertain. It is assumed to follow a Gaussian distribution with mean 'quality-mean' and standard deviation 'quality-sd'. Each time a student purchases an item, the quality of the item is sampled from this Gaussian distribution.

3. The ‘budget’ box represents the maximum amount of money that a buyer can spend in the item.

Finally, the instructor has the option to activate an information network. When activated, the students will be able to see the quality of the items purchased by the classmates directly connected to them in the network. In particular,the students are randomly placed at the nodes of a regular ring lattice of N (number of students) nodes and even ‘num-neighbors’ degree (i.e number of links). The instructor always see the complete network in her interface, but each student can only see her neighborhood.

Instructor control panel (3)



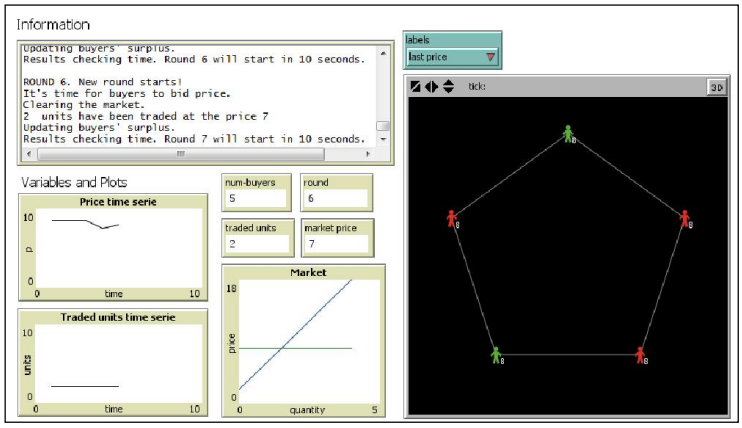
Once all the students have connected to the server, the instructor must click on the SETUP TRADE button. This will create a virtual market according to the parameters chosen by the instructor. When the instructor is ready to start the session, she may click on the GO TRADE button. The first round will start.

The experiment consists of several rounds, and each round is divided into two phases (i.e. trading-time and after-trading-time). During the trading-time, the students have the possibility of bidding (i.e. if a student is interested in an item, she has to post the price she is willing to pay at this time). The instructor can modify the duration of this phase by typing the desired duration in the box entitled ‘trading-time’. When the first phase ends, the second phase will start automatically. In this phase, the system calculates the market clearing price and the student’s interface shows her benefit in the current round, which depends on the price she proposed during the first phase, the market clearing price and the item quality (see Eq.1).

$$'Surplus' = \begin{cases} 'Budget' + 'Item\ quality' - 'Market\ price' & (if\ buyer\ purchases) \\ 'Budget' & (otherwise) \end{cases} \quad \text{(Eq.1)}$$

During this phase, the students may assess the results of the current round and maybe reflect on a strategy for the next rounds. The duration of this phase can also be modified by the instructor by typing the desired duration in the box named ‘after-trading-time’. Notice that both durations (trading and after-trading times) are measured in seconds.

Information (4)



The purpose of this part of the interface is to provide the instructor with useful information during the execution of the experiment.

The information shown in the upper left display seeks to help the instructor to manage the students along the experiment. The instructor may use the information displayed in this

window to remind the students to do a particular thing at a particular time (e.g. when to bid, when to check the results, etc).

Under the display, there are several information boxes: current round, number of buyers (i.e. the number of students that are currently connected to the server), number of units that were traded during the last round and market clearing price. The two plots on the left show the evolution of the market over the last few rounds: prices and traded units. There is another plot that shows the intersection of the supply function (defined by the instructor) and the demand function (which depends on the number and the value of the students' bids), which yields to the market clearing price. Once the experiment is over, the instructor and the students may discuss the data of these graphs.

The panel on the right displays a window with the buyers (students) that are taking place in the experiment. If the network switch is activated, the instructor will see who each student's neighbours are. If the network switch is not activated, the students will be represented as isolated agents. The color of each agent depends on whether or not the student purchased an item in the previous round (green / red respectively). The label selector on top of this black window allows the instructor to choose what information to display next to each agent (last price, last quality or none).

EXPLORING THE STUDENT'S INTERFACE

This interface provides information to help the student to decide whether or not to buy an item and, if so, what price to bid (Figure 10).

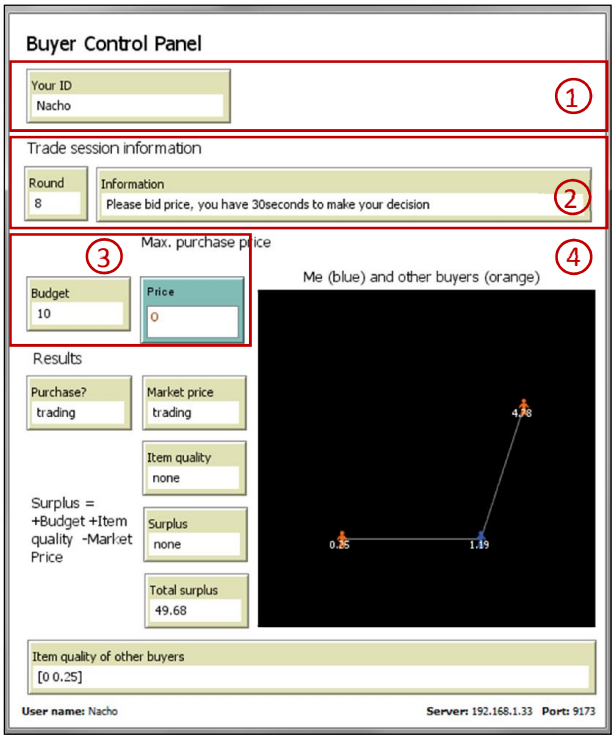
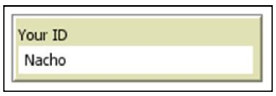


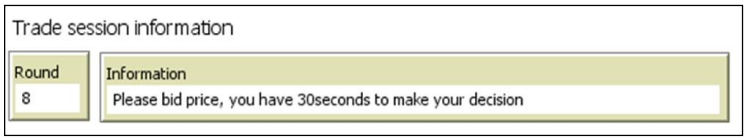
Figure 10. Student's interface

ID information (1)



This box displays the student's ID. It shows the name introduced by the student when she connects to the server (Figure 6).

Trade session information (2)

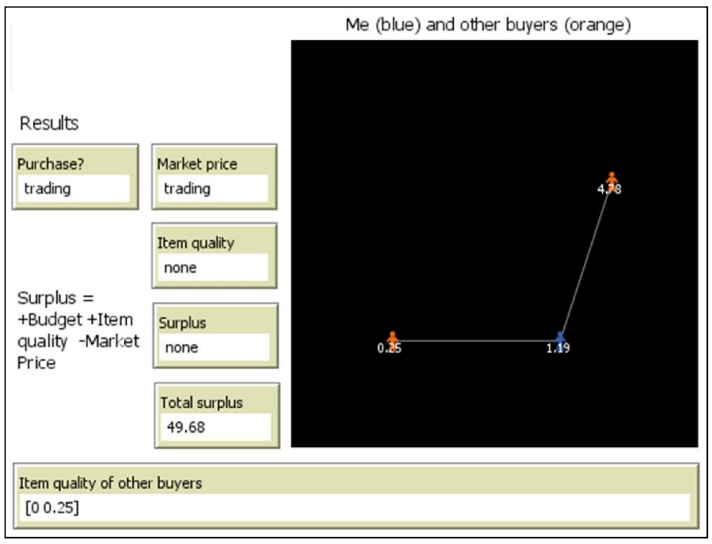


This section shows the current round and reminds the student what she is expected to do at every step along the experiment.

Maximum purchase price (3)

This section consists of two boxes: the one on the left ('Budget') is the maximum amount of money a student can spend in an item. This value is the same for all the students and is defined by the instructor. The one on the right ('Price') allows the student to input a bid for the item she wants to buy. **The student must press return after typing any bid**.

Results information (4)



The window on the right shows several agents representing the student (in blue) and her neighbors (in orange) Depending on the instructor's decision, a number may appear next to each agent (price or quality in the previous round).

The information panels on the left indicate whether or not the student has bought an item. If so, the market price, the item quality and the surplus will be displayed. The surplus is calculated as the difference between the student's budget minus the market clearing price plus the resulting quality of the item. The total surplus is the sum of the cumulative surpluses along the experiment.

The box entitled 'Item quality of other buyers' will display a vector with the quality of the items acquired by the student's neighbors in her social network.

REFERENCES

Akerlof, George A. (1970) The market for lemons: quality uncertainty and the market mechanism. *Quarterly Journal of Economics* 84:488–500

Izquierdo, Segismundo S., Izquierdo, Luis R. (2007) The impact of quality uncertainty without asymmetric information on market efficiency. Journal of Business Research 60: 858-867

Wilensky, Uri (1999). NetLogo (and NetLogo User Manual). Center for Connected Learning and Computer-Based Modeling, Northwestern University. http://ccl.northwestern.edu/netlogo/