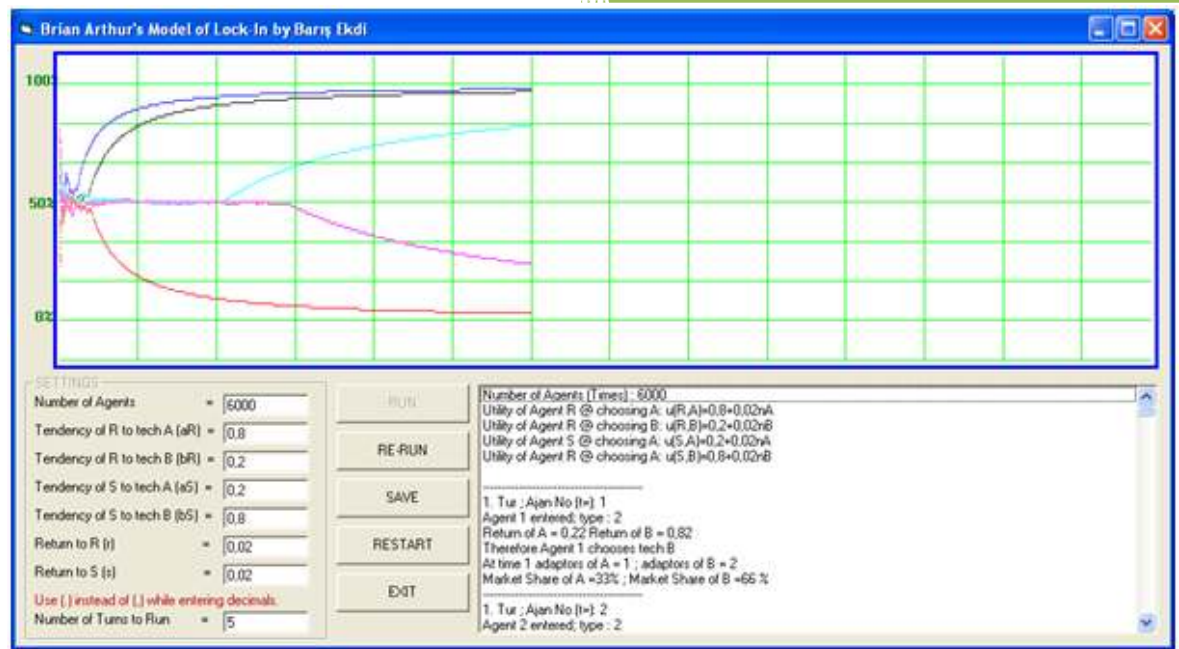


2007

Brian Arthur Model on Competing Technologies



Barış Ekdi

4/13/2007

Brian Arthur Model on Competing Technologies, Increasing Returns and Lock-In by Historical Events

METU STPS 514, Assignment II

By Barış EKDi

1- The Model of Brian Arthur

Arthur generalized his model for the purposes of studying economic processes like standardization, network effects, and so-called ‘increasing returns’. In this model, we can assume that the choices made by previous adopters from among competing technologies will matter for individual consumers.

In his model there are two type agents, namely R and S with ‘natural’ inclination to one of the technologies, A or B, respectively (as a_R, b_R, a_S, b_S in the model). However, apart from this inclination the agent also takes the returns (and network effects) of those competing technologies while making his/her choice. Therefore the number of previous adaptors chosen one of the technologies does matter. So, the model is as follows:

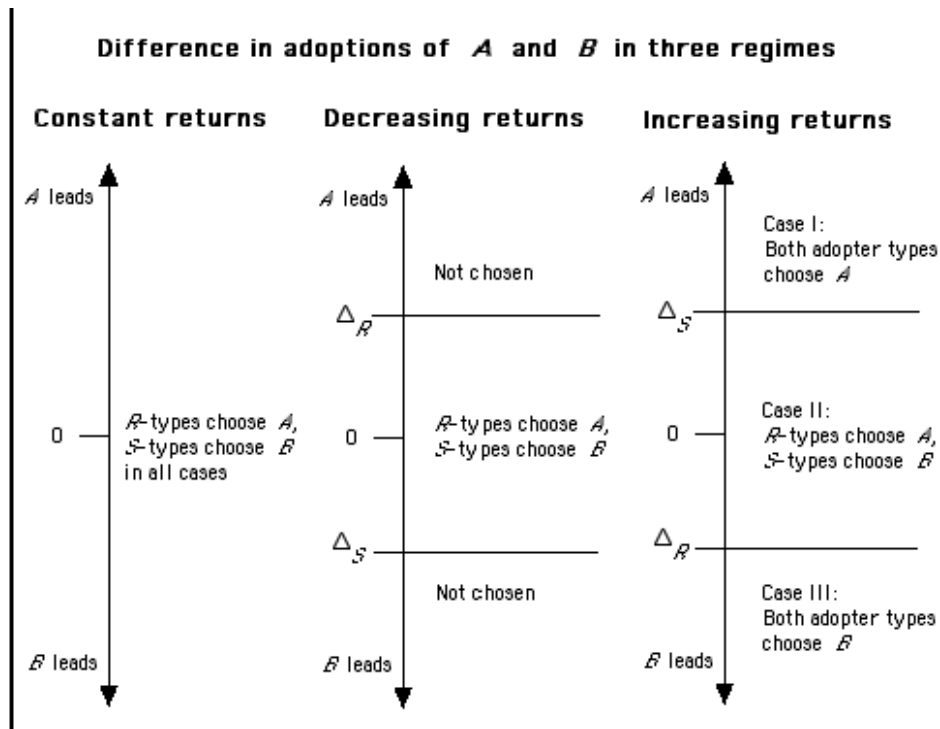
	Technology A	Technology B
Agent R	$a_R + rn_A$	$b_R + rn_B$
Agent S	$a_S + sn_A$	$b_S + sn_B$
Returns to adopting A or B, given n_A and n_B previous adopters of A and B. (The model assumes that $a_R > b_R$ and $b_S > a_S$. Both r and s are positive.)		

According to the model, R and S type agents enter the market randomly and make their choice according to the returns of competing technologies.

Arthur assumes that there are three regimes based on (i) constant returns, (ii) decreasing returns and (iii) increasing returns and those regimes result in different outcomes:

	Predictable	Flexible	Path-dependent	Path-efficient
Constant Returns	YES	NO	NO	YES
Diminishing Returns	YES	YES	NO	YES
Increasing Returns	NO	NO	YES	NO

Those predictions can be summarized as follows:



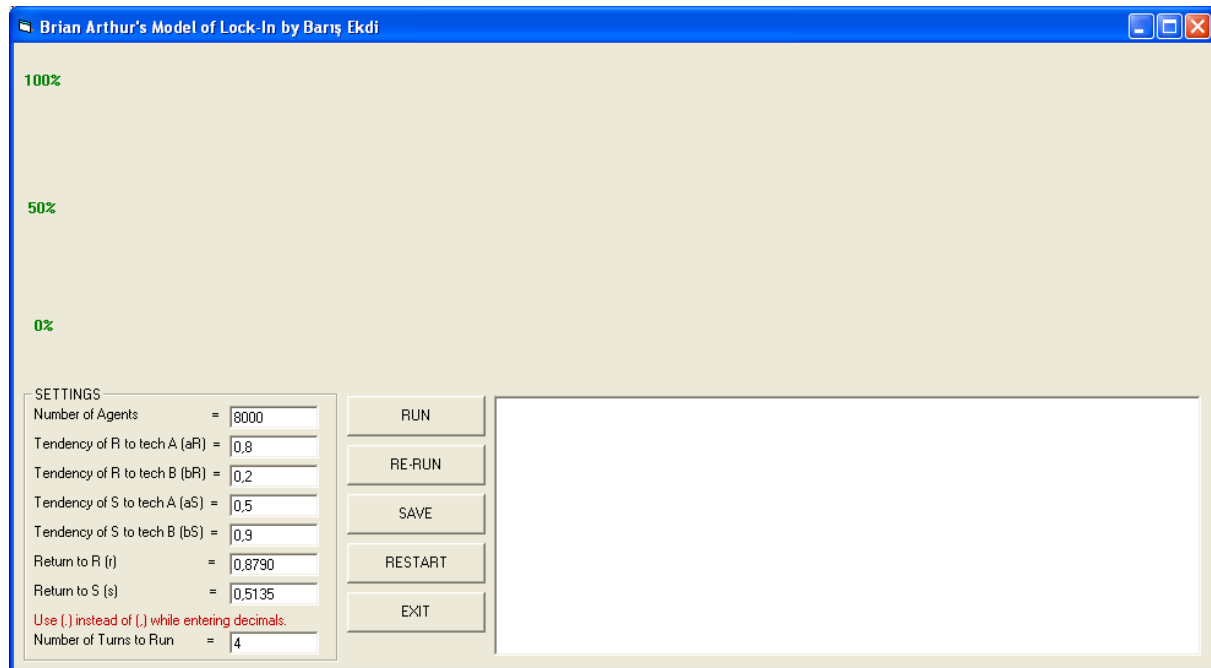
Source: Esben Sloth ANDERSEN, 2001

Therefore, the market shares of competing technologies are analyzed through a simulation in this work.

2- The Algorithm of the Program

The program “*Brian Arthur's Model of Lock-In*” is written in MS Visual Basic 6.0; and the works of Leydesdorff & Besselaar and a working paper by Andersen (2001) is used to clarify the model.

In the program a GUI (graphical user interface) is used to collect the parameters, run the simulation and show the outcome on a plot. A screenshot of the GUI is:

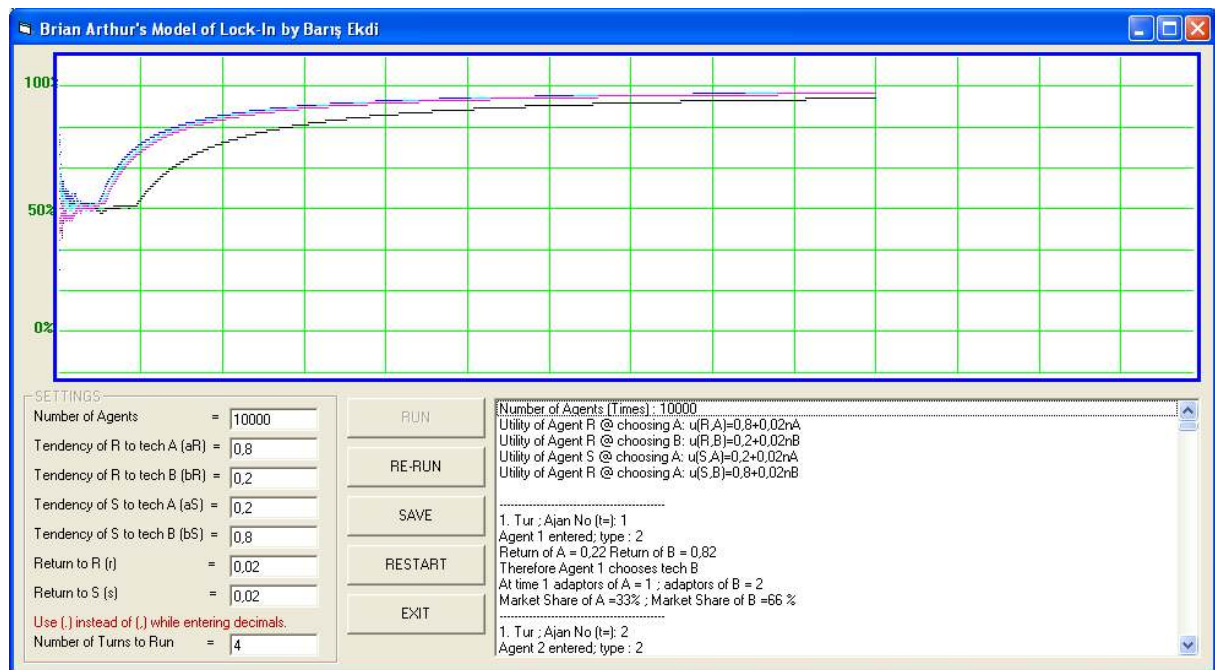


The algorithm and processes of the program is briefly listed below:

- i. The user sets the parameters or uses the parameters randomly set while the program is being loaded.
 - *Number of Agents* represents the total number of agents R and S; (and also the number of the times the market access will occur.)
 - *Tendency of R to tech A* ... and following three boxes are used to set the parameters : a_R , b_R , b_S , a_S .
 - *Return to R* and *Return to S* boxes are used to set the parameters 'r' and 's' in the model.
 - The user can also set the number of turn the simulation will run; and also continue running afterwards while pushing <RE-RUN> button.
- ii. By pressing <RUN> button the simulation starts:
 - For the number of times set in *Number of Turns to Run* box;
 - For each agent until the total number of agents is reached...

- i. A coin is tossed [rnd*1] whether the agent entering the market is type R or type S.
 - ii. The returns of this agent (R or S) provided by choosing technology A and technology B are calculated according to the formula given in the model, and parameters set by the user.
 - iii. Returns are compared and the agent chooses the technology providing higher returns.
 - iv. Number of agents using the technology chosen increases by one, thus the market share of this technology.
 - v. Results of (i) to (iv) are also printed in the box at the southeast corner of the window; for transparency of the program and further inspection of the process.
 - vi. The market share of A is represented on the graph on the top part of the window whenever an agent makes its choice. [This area for the graphics is suitable for allocating up to 13900 agents]
- Processes i – vi repeated for each agent until all the agents make their choices.
 - This part is repeated until the number of turns set by the user is fulfilled. Outcome of each turn represented on the graph with 6 different colors.
- iii. After this process user is free to continue the simulation by pressing <RE-RUN> button, or <SAVE> the process log or <RESTART> another simulation by pressing the relevant button.

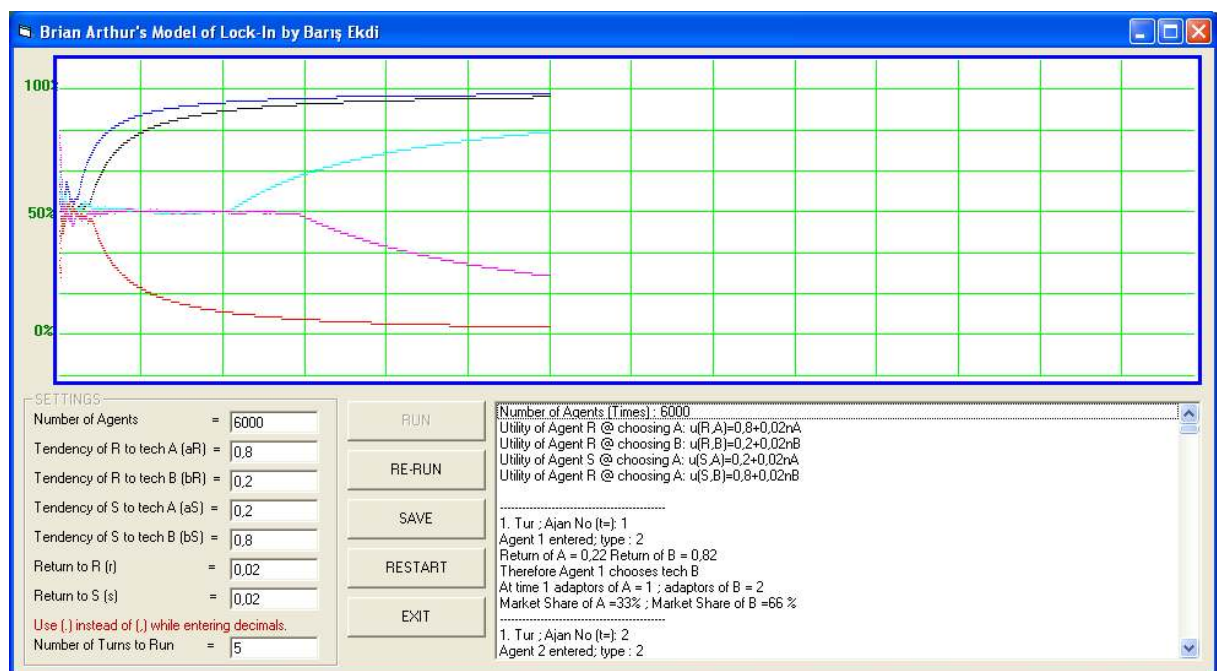
There is no inbuilt tool available for copying the graphical outcome yet, however the user can copy the screenshot by pressing <Alt+PrtScr> to the clipboard and paste it into a document. Outcome of a simulation run 4 times with 10.000 agents is below:



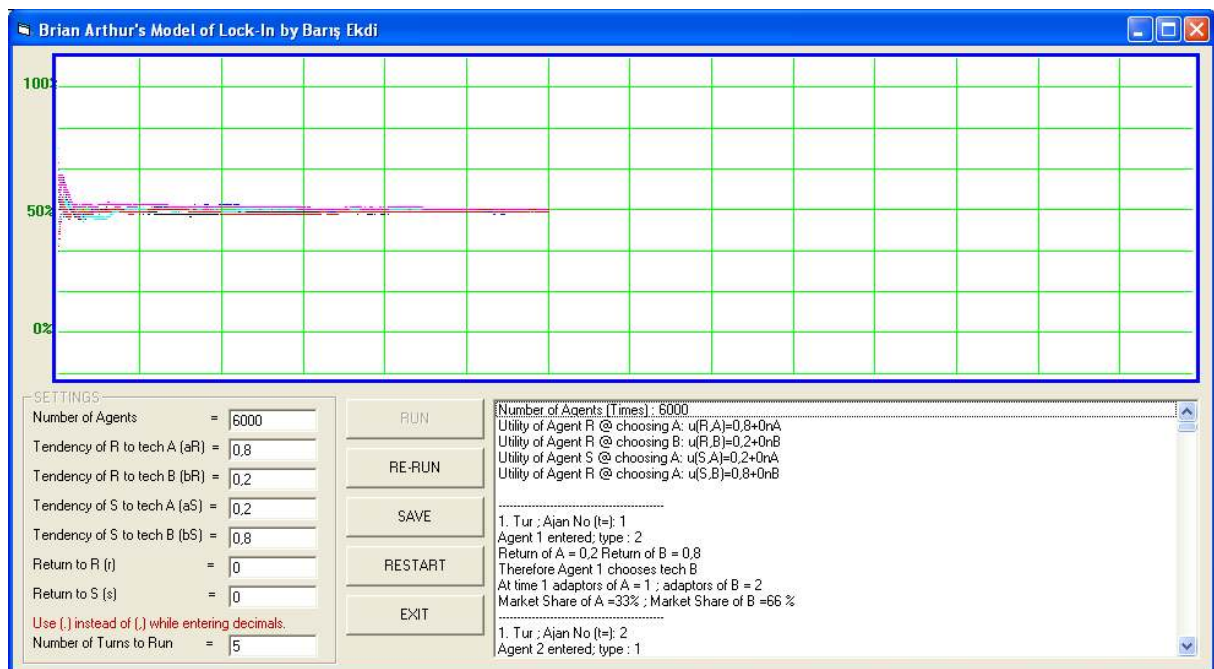
3- Outcomes of the Simulations

Predictions of Arthur is tested on the program:

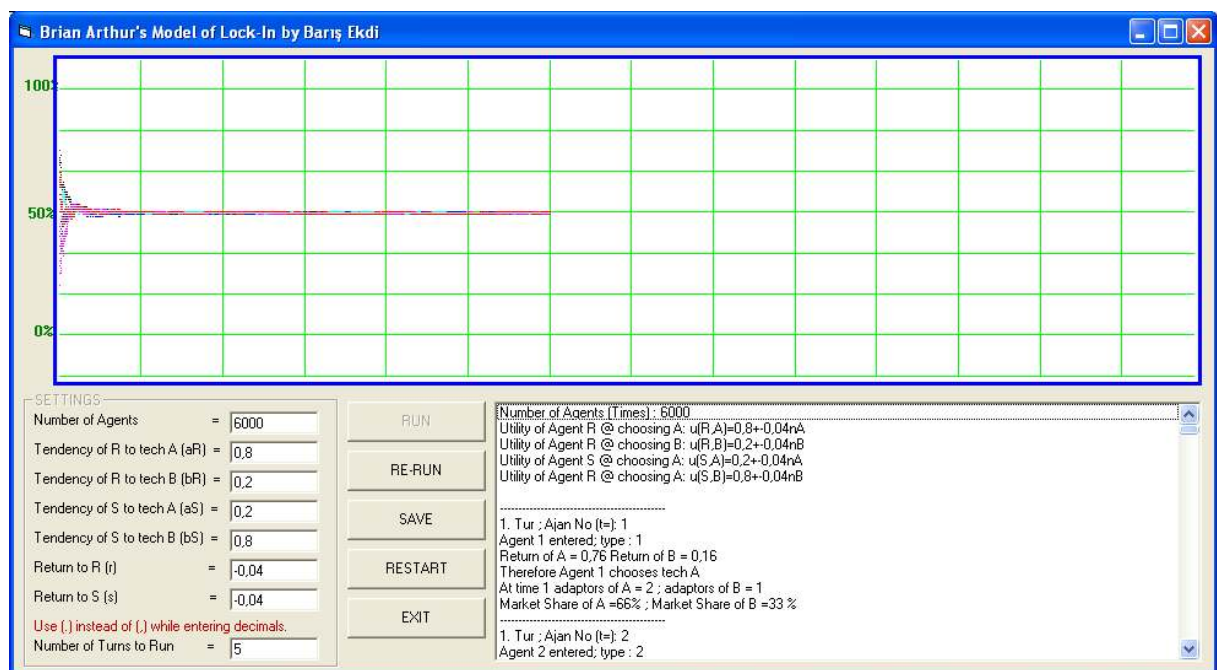
- A set with **increasing returns** ($r,s=0,02$ for both R and S) , and with 6000 agents run 5 times. Lock-in occurred after 1000th agent. The result is as follows:



- A set with **constant returns** ($r,s=0$ for both R and S) , and with 6000 agents runs 5 times. Both technologies almost have equal shares. The result is as follows:



- A set with **diminishing returns** ($r,s = -0,04$ for both R and S) , and with 6000 agents runs 5 times. The result is as follows:



- All results are consistent with the predictions set in the table at the 2nd page of this work.