Bounded Rationality and Learning
in
Dynamic Programming Environments

A THESIS
SUBMITTED TO THE DEPARTMENT OF ECONOMICS
AND THE INSTITUTE OF ECONOMICS AND SOCIAL SCIENCES OF
BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

MASTER OF ARTS IN ECONOMICS

By

Mahmut Erdem

February, 2001

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Arts.

Asst. Prof. Dr. Erdem Baþçý (Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Arts.

Assoc. Prof. Dr. Azer Kerimov

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Arts.

Assoc. Prof. Dr. Farhad Husseinov

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Arts.

Asst. Prof. Dr. Nedim Alemdar

Approved for the Institute of Economics and Social Sciences:

Prof. Dr. Kürþat Aydoðan
Director of the Institute of Economics and Social Sciences

ABSTRACT

BOUNDED RATIONALITY AND LEARNING
IN DYNAMIC PROGRAMMING ENVIRONMENTS

Mahmut Erdem
M.A in Economics
Supervisor: Asst. Prof. Dr. Erdem Başçı
February 2001

The purpose of this thesis is to explain "excess sensitivity" puzzle observed in consumption behavior an alternative way. By deviating from full optimization axiom, in a dynamic extension of Arthur's stochastic decision model, it was observed that a tendency of excess consumption following temporary income shock prevails. Another main technical contribution achieved in this thesis is in modelling behavior and learning in intertemporal decision problems. In particular, an extension of Arthur's type of behavior to dynamic situations and comparison of the corresponding values with those of Bellman's dynamic programming solution is achieved. Moreover it was shown by using stochastic approximation theory that classifier systems learning ends up at the 'strength' values corresponding to the Arthur's value function.

Keywords: Dynamic programming, value function, classifier systems learning, stochastic approximation theory, excess sensitivity puzzle, consumption.

ÖZET

DİNAMİK PROGRAMLAMA ORTAMLARINDA
SINIRLI RASYONELLİK VE ÖĞRENME

Mahmut Erdem
İktisat Bölümü, Yüksek Lisans
Tez Yöneticisi: Yrd. Doç. Dr. Erdem Başçı
Şubat 2001

Bu çalışma, tüketim davranışlarında gözlenen "aşırı duyarlılık" problemine farklı bir açıklama getirmeyi amaçlamaktadır. Optimizasyon varsayımından ayrılmakla, Arthur'un stokastik davranış modelinin dinamik bir uzantısında, anlık gelir şoklarının fazla tüketme eğilimine yolaçtığı gözlenmistir. Bir başka teknik katkı da, dinamik ortamlarda davranış biçimlerini ve öğrenmeyi modellemedir. Arthur'un öngördüğü davranış biçiminin dinamik programlama problemlerinde aldığı değerlerin, Bellman eşitliğinin çözümüyle kıyaslaması da yapılmıştır. Suni zeka literatüründe önerilmiş olan sınıflandırıcı sistemler, tüketim probleminde öğrenme modeli olarak kullanılmıştır. Sınıflandırıcıların limitteki güçleri ile Arthur'un değerleri arasında bir denklik olduğu gösterilmistir.

Anahtar Sözcükler: Dinamik programlama, değer fonksiyonu, sınıflandırıcı sistemlerle öğrenme, stokastik yaklaşım, aşırı duyarlılık problemi, tüketim.

ACKNOWLEDGEMENTS

I would like to express my gratitude to Asst. Prof. Dr. Erdem Başçı for drawing my attention to the subject, for providing me the necessary background I needed to complete this study and the valuable supervision he provided. I also would like to thank Assoc. Prof. Dr. Azer Kerimov, Prof. Dr. Vladimir Anisimov, Assoc. Prof. Dr. Ferhat Hüsseinov and Asst. Prof. Dr. Nedim Alemdar for their valuable comments concerning this study.

It would have been really difficult for me to submit a proper thesis on time without the help of Uğur Madran. The help he offered while I was trying to solve equaitions with Maple is gratefully acknowledged.

To Orhan Güvenen

Contents

Chapter 1: Introduction

In dynamic economic models agents are assumed to behave like their decisions are the solution of the dynamic programming problem. A great deal of research effort has been devoted to support this paradigm with observations. Although this effort led to many successful explanations , it also met some puzzles (see John Rust, 1992). Thus, many of the researchers have studied on alternative ways of defining the human rationality, mostly under the subject of ‘bounded rationality’ (see Sargent , 1993 and Simon 1982).

As a model of consumption-savings behavior, the permanent-income hypothesis (PIH) has occupied a central position in macroeconomics since Milton Friedman (1957). Although the PIH is taken as axiomatic in many macroeconomic studies, its empirical accuracy is questioned in the current empirical literature. Many of these studies find that consumption growth rates are positively correlated with predictable changes in real income. This finding is sometimes described as “excess sensitivity” of consumption to income and interpreted as strong evidence against the PIH. In an offered explanation, rule-based decision-making, the decisions generally differ with that of dynamic programming solution even if one of the decision rules is in the dynamic programming solution.(Lettau and Uhlig, 1999)

The ‘rule of thumb’ is a remarkable type of learning model which was studied by economists including, Ingram (1990), Campbell and Mankiw (1990), Binmore and Samuelson (1992), Lusardi (1996). Learning takes place by evaluating the quality of competing rules of thumb via past experiences from using them, using a simple updating algorithm. Lettau and Uhlig (1999) explains the ‘excess sensitivity’ by showing that agents can learn ‘falsely’ suboptimal rules to some others implementing the ‘optimal’ decisions. The main reason for this to happen is the agents indifference between ‘smart behavior’ and ‘good luck’. The main argument is that (Lettau and Uhlig, 1999, pp: 169) “.. bad decisions in good times ‘feel better’ than good decisions in bad times’. The learning scheme investigated in Lettau and Uhlig (1999) gives rise to a “good state bias,” , i.e it favors bad decisions applicable only in good states. The explanation of ‘excess sensitivity’ by the feature of ‘good state bias’ may help in resolving the puzzle pointed out by Flavin(1981), Zeldes (1989).

In this dissertation, we suggest an alternative explanation to this puzzle. By deviating from full optimization axiom, we refer to the stochastic decision model suggested by Arthur (1989). This model views agents as behaving according to the relative perceived payoffs of alternative strategies. In a consumption framework, which is dynamic, we extend Arthur’s model and observe that a tendency of excess consumption following temporary income shocks prevails. The main reason is that overconsumption is better than underconsumption , although, both are inferior to optimal consumption.

Another main technical contribution of this thesis is in modelling behaviour and learning in intertemporal decision problems. In particular, an extension of Arthur’s (1991) type of behavior to dynamic situations and comparison of the corresponding values with those of Bellman’s dynamic programming solution is achieved. Moreover, we study the dynamics of learning. Using stochastic approximation theory (Ljung, 1977), we show that classifier systems learning, (see Holland, 1975), ends up at the ‘strength’ values corresponding to the *Arthur’s value function.* We are heavily influenced by Lettau andd Uhlig (1999) and Metivier and Priouret (1984). Shortly saying, our learning model is of the Ljung’s (1977) type and satisfies certain sort of continuity conditions. The Theorem (see Appendix A) implies the existence of limit point(s) and convergence.

The organization of the thesis is as follows: Chapter 2, presents the Bellman equation and the numerical solution under some assumptions. Also a new function is introduced, namely, *Arthur’s value function* and corresponding *augmented value function* is defined. Chapter 3 introduces the ‘learning with past experiences’, and the ‘convergence’ concept of the strengths. In Chapter 4, we explain the simulation results.

Chapter 2: Dynamic Programming

2.1 Bellman equation

First we will begin by studying the well-known cake-eating problem to find the optimal values for the consumer's possible different consumption decisions. We assume that for the case of learnability, the consumer has a probability of getting new subsidy , $p_s \in (0,1)$. This makes the dynamic optimization problem a repeated one.

Now let's describe our framework in general terms. Time is discrete, i.e $t \in \mathrm{N}$, and the consumer is infinitely lived. At time *t* consumer has k_t amount of cake from the set $\mathrm{X} = \{0,1,...,\bar{k}\}$, the state space, and is allowed to consume $0 \leq c_t \leq k_t$ amount of cake. The consumer then experiences the instantaneous utility $u(c_t) \in R$ and the new state k_{t+1} according to the some probability distribution which will be described later. The total time-zero expected utility is given

$$U_0 = E_0 \sum_{t=0}^{\infty} \beta^t u(c_t),$$

where $0 < \beta < 1$, is a discount factor and E_0 is the conditional expectations operator. Most recursive stochastic dynamic decision problems can be formulated in this way at least approximately by discretizing the state space and the action space by changing zero transition probabilities to some small non-negative amount.

The variable that makes this problem stochastic is the subsidy that was mentioned before in the previous paragraphs. If the consumer has 0 cake in hand at the end of period *t*, government , with probability p_s, serve an amount of $\bar{k}$ cakes at the beginning of time *t+1*.

Now, we can write the following Bellman's equation for this dynamic optimization problem:

$$v(k) = \max\{u(c) + \beta E v(k - c + s) | c \in \mathrm{X}, c \leq k\} \qquad (1)$$

for all $k \in \mathrm{X}$, and s is the amount of the subsidy which is $\bar{k}$ with probability p_s, and is 0 with probability 1- p_s. Here, $v : \mathrm{X} \to R$, which is called *optimal value function,* gives the maximum lifetime expected utility from having k units of cake in hand. This equation can be solved by using iteration on $v(k)$'s but we will use a different, simple method.

From now on, we will assume that $\mathrm{X} = \{0,1,2\}$, and $\bar{k} = 2$. For these 3 k values, let's write the Bellman's equation explicitly:

$$v(2) = \max\{u(2) + \beta p_s v(2) + \beta(1 - p_s)v(0), u(1) + \beta v(1), u(0) + \beta v(2)\} \qquad (2)$$

$$v(1) = \max\{u(1) + \beta p_s v(2) + \beta(1 - p_s)v(0), u(0) + \beta v(1)\} \qquad (3)$$

$$v(0) = \{u(0) + \beta p_s v(2) + \beta(1 - p_s)v(0)\}. \qquad (4)$$

the 3'rd term in the equation (2) and the 2'nd in (3) are dominated, hence can be ignored. Say, for simplicity, $v_2 = v(2), v_1 = v(1), v_0 = v(0)$ are the solutions of equations (2),(3),(4). Solving (4) in terms of v_2 and similarly solving (3) in terms of v_2 gives us two equations:

$$v_0 = \frac{\beta p_s}{1 - \beta(1 - p_s)} v_2 \text{........(5)}, \quad v_1 = 8 + \beta p_s v_2 + \frac{\beta(1 - p_s)\beta p_s}{1 - \beta(1 - p_s)} v_2 \text{....(6)}$$

and finally substituting the two equations into (2) gives us the equation:

$$v_2 = \max\left\{u(2) + v_2\left[\beta p_s + \frac{\beta(1-p_s)\beta p_s}{1-\beta(1-p_s)}\right], u(1) + u(1)\beta + v_2\left[\beta^2 p_s + \frac{\beta^3(1-p_s)p_s}{1-\beta(1-p_s)}\right]\right\}$$

This equation in terms of the single unknown v_2 can be solved under a given set of parameter values $\beta, p_s, u(0), u(1), u(2)$.

Also, there exists another way to find the solution of above 3 equations; defining a contraction mapping T on continuous real valued functions with domain X. It is also easy to verify that this mapping satisfies the Blackwell's sufficient conditions for a contraction. (For details the reader is referred to Stokey and Lucas (1989)).

Now, we define another function, *augmented value function* for this dynamic optimization problem. This function represented by $\bar{v}$, operates on $\mathrm{X} \times \mathrm{X}$ and maps to the real line, R. Its interpretation is that $\bar{v}(k, c)$ gives the lifetime expected utility from having initial cake size of k and consuming c in the first period, but following optiumal policies thereafter.

$$\bar{v}(k,c) = u(c) + \beta E v(k - c + s) \text{ for all } (k,c) \in X \times X .$$

For the case of $k \in \mathrm{X} = \{0,1,2\}$, we can write the 6 corresponding *augmented values*;

$$\bar{v}(2,2) = u(2) + \beta p_s v(2) + \beta(1-p_s)v(0),$$

$$\bar{v}(2,1) = u(1) + \beta v(1),$$

$$\bar{v}(2,0) = u(0) + \beta v(2),$$

$$\bar{v}(1,1) = u(1) + \beta p_s v(2) + \beta(1-p_s)v(0),$$

$$\bar{v}(1,0) = u(0) + \beta v(1),$$

$$\bar{v}(0,0) = u(0) + \beta p_s v(2) + \beta(1-p_s)v(0),$$

To understand the behaviour of the solution we now give numbers to the parameters in the former equations. Let, $u(0)=0, u(1)=8, u(2)=10$ and the discount factor $\beta=0.9$, $p_s \in \{0.2,0.3,0.4,0.5,0.6\}$. Now let us write down the optimal values for this set of probabilities and compare them.

	$v(2)$	$v(1)$	$v(0)$
$p_s=0.2$	28	26	18
$p_s=0.3$	44.28	40.31	32.31
$p_s=0.4$	51.41	48.23	40.23
$p_s=0.5$	57.65	55.16	47.16
$p_s=0.6$	64	62	54

and the corresponding $\bar{v}(.,.)$ values are;

	$\bar{v}(2,2)$	$\bar{v}(2,1)$	$\bar{v}(2,0)$	$\bar{v}(1,1)$	$\bar{v}(1,0)$	$\bar{v}(0,0)$
$p_s=0.2$	28	31.4	25.2	26	23.4	18
$p_s=0.3$	42.31	44.27	40.31	40.31	36.28	32.31
$p_s=0.4$	50.23	51.41	46.27	48.23	43.41	40.23
$p_s=0.5$	57.16	57.64	51.88	55.16	49.64	47.16
$p_s=0.6$	64	63.8	57.6	62	55.8	54

For the $p_s \in \{0.2,0.3,0.4,0.5,0.6\}$, except p_s=0,6 , $\bar{v}(2,1)$ is the highest among all others. And for all p_s, $\bar{v}(1,1)$ is greater or equal to *v(k,0)*. Another main point is, if p_s=0.6,

$\bar{v}(2,2) > \bar{v}(2,1)$. That is, as the p_s increases the consumption pattern of the consumer switches to mode of consuming more.

2.2 Arthur's value function

In this section we will deal with a similar problem that was analyzed in the last section. The main difference is now in the new value function , call it *Arthur's value function*, which does not give the maximal amount of expected lifetime utility , but it gives the expected amount of lifetime utility attainable by the consumer who begins with a specified amount of cake in hand and follows the behavior suggested by Arthur (1989). The assumptions that we made in the last section , except this behavioral assumption , will be valid in this section too. We will denote the new value function as $v_r : X \to R$, i.e,

$$v_r(k) = E_{c,s}\left\{u(c) + \beta v_r(k-c+s) \middle| c \in \mathrm{X}, c \leq k\right\} \tag{7}$$

and the corresponding *Arthur's augmented value function* $\bar{v}_r : X \times X \to R$ as follows:

$$\bar{v}_r(k,c) = u(c) + \beta E_s v_r(k-c+s)\,,\ \forall k \in X, c \leq k\,.$$

Arthur's behavioral model suggests that consumer's likelihood of choosing an action is proportional to its payoff. In our dynamic setup, therefore we will define the probabilities as follows:

$$P(c,k,v_r) = \frac{\bar{v}_r(k,c)}{\sum_{c \leq k} \bar{v}_r(k,c)}\,, \text{ where } \quad \bar{v}_r(k,c) = u(c) + \beta E_s v_r(k-c+s)$$

In order to find the unknown function v_r, we will define a mapping $T : C(X) \to C(X)$ so that v_r will be a fixed point of this mapping. The Brouwer's fixed point theorem implies the existence of a fixed point . We will just solve the 3 equations below and find the values satisfying non-negativity constraint. First, let us write down the implied equations explicitly:

$$T(v_r)(k) = \sum_{c \le k} P(c,k,v_r)u(c) + \beta \sum_{c \le k} P(c,k,v_r) E_{s|c} v_v(k-c+s) \tag{8}$$

where,

$$P(0,0,v_r) = 1$$

$$P(0,1,v_r) = \frac{\beta v_r(1)}{\beta v_r(1) + u(1) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)}$$

$$P(1,1,v_r) = \frac{u(1) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)}{\beta v_r(1) + u(1) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)}$$

$$P(0,2,v_r) = \frac{\beta v_r(2)}{\beta v_r(2) + u(1) + \beta v_r(1) + u(2) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)}$$

$$P(1,2,v_r) = \frac{u(1) + \beta v_r(1)}{\beta v_r(2) + u(1) + \beta v_r(1) + u(2) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)}$$

$$P(2,2,v_r) = \frac{u(2) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)}{\beta v_r(2) + u(1) + \beta v_r(1) + u(2) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)}$$

substituting these 6 probability values into the equation (9) and equating right hand side to corresponding Arthur's values gives us the following 3 equations;

$$T(v_r)(0) = \beta p_s v_r(2) + \beta(1-p_s)v_r(0) = v_r(0) \tag{9}$$

$$T(v_r)(1) = \frac{(\beta v_r(1))^2 + (u(1) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0))^2}{\beta v_r(1) + u(1) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)} = v_r(1) \tag{10}$$

$$T(v_r)(2) = \frac{(\beta v_r(2))^2 + (u(1) + \beta v_r(1))^2 + (u(2) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0))^2}{\beta v_r(2) + u(1) + \beta v_r(1) + u(2) + \beta p_s v_r(2) + \beta(1-p_s)v_r(0)} = v_r(2) \tag{11}$$

and the corresponding *Arthur's augmented value function is defined by ;*

$\bar{v}_r(0,0) = u(0) + \beta v_r(2) + \beta(1-p_s)v_r(0)$,

$\bar{v}_r(1,0) = u(0) + \beta v_r(1)$

$\bar{v}_r(1,1) = u(1) + \beta v_r(2) + \beta(1-p_s)v_r(0)$

$\bar{v}_r(2,0) = u(0) + \beta v_r(2)$

$\bar{v}_r(2,1) = u(1) + \beta v_r(1)$

$\bar{v}_r(2,2) = u(2) + \beta v_r(2) + \beta(1-p_s)v_r(0)$

Unfortunately we are unable to solve equations, (9),(10) and (11) by hand, but instead we wrote a simple algorithm on *Mappl*e and solved the equations simultaneously. Restricting the solutions to nonnegative real numbers gives us unique values for $v_r(k)$, $k \in X$.

	$v_r(2)$	$v_r(1)$	$v_r(0)$
$p_s = 0.2$	27.16	23.50	17.46
$p_s = 0.3$	32.86	29.52	23.98
$p_s = 0.4$	37.72	34.64	29.52
$p_s = 0.5$	41.92	39.05	34.30
$p_s = 0.6$	45.58	42.90	38.46

and the corresponding $\bar{v}_r(k,c)$ values are;

	$\bar{v}_r(2,2)$	$\bar{v}_r(2,1)$	$\bar{v}_r(2,0)$	$\bar{v}_r(1,1)$	$\bar{v}_r(1,0)$	$\bar{v}_r(0,0)$
$p_s = 0.2$	27.46	29.15	24.44	25.46	21.15	17.46
$p_s = 0.3$	33.98	34.57	29.57	31.98	26.57	23.98
$p_s = 0.4$	39.52	39.18	33.95	37.52	31.18	29.52
$p_s = 0.5$	44.30	43.15	37.73	42.30	35.15	34.30
$p_s = 0.6$	48.46	46.61	41.02	46.46	38.61	38.46

Since the agent does not maximize his or her payoffs in Arthur's behavioral model, the corresponding *Arthur's augmented values* are lower than that of optimal values. Shortly saying, $\bar{v}_r(1,1)$ is the highest among all$\{\bar{v}_r(0,0), \bar{v}_r(1,0), \bar{v}_r(2,0), \bar{v}_r(1,1)\}$, and for $p_s \geq 0.4$ $\bar{v}_r(2,2) > \bar{v}_r(2,1)$.

Chapter 3:Learning With Past Experiences

In this Chapter we will consider the learning model of an agent who does not know anything about payoffs. The agent will be assumed to have subjective beliefs on the values of each possible pair of state and action, *(k,c)* and update these values through experience.

For dynamic decision environments, Lettau and Uhlig (1999) propose a learning algorithm based on classifier systems. Classifier systems learning, introduced by Holland (1975) as a tool for machine learning, is also suitable for modeling Arthur's type of learning . A classifier systems consists of a list of *condition-action* statements, which are called classifiers, and a corresponding list of real numbers, called the *strengths* of these classifiers. Classifiers bid their strengths in competition for the right to guide the agent in each decision situation. The strengths are then updated according to the outcomes.

In our learning model there are three main steps in operation of a classifier system.

1. *Activation:* Recognize the current condition and determine the list of applicable classifiers in the current condition,
2. *Selection:* Select one of the applicable classifiers with probability equal to the weight of selected one among the others,
3. *Update:* Update the *strengths* according to an adjustment formula.

Now let us give first the preliminaries and then check whether our model satisfies the conditions of the theorem given by Metivier and Priouret (1984). The explicit form of our strength update formula is ;

$$\theta_{t+1} = \theta_t - \gamma_{t+1} f(\theta_t, Y_{t+1}) \tag{12}$$

where $f: R^d \times R^k \to R^d$ and

$$f(\theta_t, Y_{t+1}) = e_{k_t, c_t} g(\theta_t, Y_{t+1}) \tag{13}$$

where e_{k_t, c_t} is the k dimensional unit vector with a one in entry (k_t, c_t) (the notation used here is different then the usual vector notation) and zeros elsewhere, and where the scalar factor $g(\theta_t, Y_{t+1})$ is given by;

$$g(\theta_t, Y_{t+1}) = \theta_{k_t, c_t} - u(c_t) - \beta\theta_{k_{t+1}, c_{t+1}} \tag{14}$$

The first equation (12) is the standard format for stochastic approximation algorithms: the strength vector θ_t is updated, using some correction, $f(\theta_t, Y_{t+1})$, weighted with the decreasing weight γ_{t+1} and stated here for the entire vector of strengths. The second equation (13) states that this correction takes place only in one component of the strength vector, namely the strength corresponding the classifier (k_t, c_t), which was activated at date *t*. The third equation (14) states by how much that entry should be changed.

Now let us define the corresponding parameters in our stochastic approximation algorithm . Let $\theta_t = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \in R^6$ is the strength vector at time *t*, and $\theta_1 = S_{22}, \theta_2 = S_{21}, \theta_3 = S_{20}, \theta_4 = S_{11}, \theta_5 = S_{10}, \theta_6 = S_{00}$. The term S_{ij} used here is to represent the *strength* of consuming j-units of cake when i-units of cake is available. That is S_{ij}, is the strength of the activated classifier.

e_{k_t, c_t} : 6 dimensional unit vector with a one in entry (k_t, c_t) and zeros elsewhere.

Let $Y_{t+1} = \left[k_t, c_t, k'_{t+1}, c'_{t+1}\right]$ be defined as; the first two terms in the vector are the time *t* decision of the consumer and the 3rd and 4th are time *t+1*, i.e, *Y* vector totally defines the present and next step consumption decisions. The possible *Y* vectors are as follows:

$Y_1 = [2,2,2,2]$, $Y_2 = [2,2,2,1]$, $Y_3 = [2,2,2,0]$, $Y_4 = [2,2,0,0]$, $Y_5 = [2,1,1,1]$, $Y_6 = [2,1,1,0]$, $Y_7 = [2,0,2,2]$, $Y_8 = [2,0,2,1]$, $Y_9 = [2,0,2,0]$, $Y_{10} = [1,1,2,2]$, $Y_{11} = [1,1,2,1]$, $Y_{12} = [1,1,2,0]$, $Y_{13} = [1,1,0,0]$, $Y_{14} = [1,0,1,0]$, $Y_{15} = [1,0,1,1]$, $Y_{16} = [0,0,2,2]$, $Y_{17} = [0,0,2,1]$, $Y_{18} = [0,0,2,0]$, $Y_{19} = [0,0,0,0]$.

Let Π_θ be the transition matrix of vector *Y*'s. That is, $\Pi_{\theta,ij} = P\{Y_j | Y_i\}$. It can be seen that $\Pi_{\theta,ij} = 0$ for the case when the 3rd and 4th of *old* vector $Y_i = [k, c, k', c']$ does not match with 1st and 2nd of *new* vector $Y_j = [\bar{k}, \bar{c}, \bar{k}', \bar{c}']$ respectively. i.e $\Pi_{\theta,ij} = 0$ if $c' \neq \bar{c}$ or $k' \neq \bar{k}$.

Now we can write the desired transition probability matrix Π_θ.

$$\Pi_{\theta,1,1} = p_s \frac{\theta_1}{\sum \theta_1 + \theta_2 + \theta_3}, \ \Pi_{\theta,1,2} = p_s \frac{\theta_2}{\sum \theta_1 + \theta_2 + \theta_3}, \ \Pi_{\theta,1,3} = p_s \frac{\theta_3}{\sum \theta_1 + \theta_2 + \theta_3},$$

$$\Pi_{\theta,1,4} = (1 - p_s), \ \Pi_{\theta,2,5} = \frac{\theta_4}{\sum \theta_4 + \theta_5}, \ \Pi_{\theta,2,6} = \frac{\theta_5}{\sum \theta_4 + \theta_5}$$

$$\Pi_{\theta,3,7} = \frac{\theta_1}{\sum \theta_1 + \theta_2 + \theta_3}, \ \Pi_{\theta,3,8} = \frac{\theta_2}{\sum \theta_1 + \theta_2 + \theta_3}, \ \Pi_{\theta,3,9} = \frac{\theta_3}{\sum \theta_1 + \theta_2 + \theta_3}$$

$$\Pi_{\theta,4,16} = p_s \frac{\theta_1}{\sum \theta_1 + \theta_2 + \theta_3}, \ \Pi_{\theta,4,17} = p_s \frac{\theta_2}{\sum \theta_1 + \theta_2 + \theta_3}, \ \Pi_{\theta,4,18} = p_s \frac{\theta_3}{\sum \theta_1 + \theta_2 + \theta_3},$$

$$\Pi_{\theta,4,19}=(1-p_s)\,,\Pi_{\theta,5,10}=p_s\frac{\theta_1}{\sum\theta_1+\theta_2+\theta_3}\,,\Pi_{\theta,5,11}=p_s\frac{\theta_2}{\sum\theta_1+\theta_2+\theta_3}\,,$$

$$\Pi_{\theta,5,12}=p_s\frac{\theta_2}{\sum\theta_1+\theta_2+\theta_3}\,,\Pi_{\theta,5,13}=(1-p_s)\,.\,\Pi_{\theta,6,14}=\frac{\theta_5}{\sum\theta_4+\theta_5}\,,\ \Pi_{\theta,6,15}=\frac{\theta_4}{\sum\theta_4+\theta_5}\,,$$

$$\Pi_{\theta,7,1}=p_s\frac{\theta_1}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,7,2}=p_s\frac{\theta_2}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,7,3}=p_s\frac{\theta_3}{\sum\theta_1+\theta_2+\theta_3}\,,$$

$$\Pi_{\theta,7,4}=(1-p_s)\,,\quad \Pi_{\theta,8,5}=\frac{\theta_4}{\sum\theta_4+\theta_5}\,,\ \Pi_{\theta,8,6}=\frac{\theta_5}{\sum\theta_4+\theta_5}$$

$$\Pi_{\theta,9,7}=\frac{\theta_1}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,9,8}=\frac{\theta_2}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,9,9}=\frac{\theta_3}{\sum\theta_1+\theta_2+\theta_3}$$

$$\Pi_{\theta,10,1}=p_s\frac{\theta_1}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,10,2}=p_s\frac{\theta_2}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,10,3}=p_s\frac{\theta_3}{\sum\theta_1+\theta_2+\theta_3}\,,$$

$$\Pi_{\theta,10,4}=(1-p_s)\,,\Pi_{\theta,11,5}=\frac{\theta_4}{\sum\theta_4+\theta_5}\,,\ \Pi_{\theta,11,6}=\frac{\theta_5}{\sum\theta_4+\theta_5}$$

$$\Pi_{\theta,12,7}=\frac{\theta_1}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,12,8}=\frac{\theta_2}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,13,9}=\frac{\theta_3}{\sum\theta_1+\theta_2+\theta_3}$$

$$\Pi_{\theta,13,16}=p_s\frac{\theta_1}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,13,17}=p_s\frac{\theta_2}{\sum\theta_1+\theta_2+\theta_3}\,,\ \Pi_{\theta,13,18}=p_s\frac{\theta_3}{\sum\theta_1+\theta_2+\theta_3}\,,$$

$$\Pi_{\theta,13,19}=(1-p_s)\,,\Pi_{\theta,14,14}=\frac{\theta_5}{\sum\theta_4+\theta_5}\,,\ \Pi_{\theta,14,15}=\frac{\theta_4}{\sum\theta_4+\theta_5}$$

$$\Pi_{\theta,15,10} = p_s \frac{\theta_1}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,15,11} = p_s \frac{\theta_2}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,15,12} = p_s \frac{\theta_3}{\sum \theta_1 + \theta_2 + \theta_3},$$

$$\Pi_{\theta,15,13} = (1 - p_s),$$

$$\Pi_{\theta,16,1} = p_s \frac{\theta_1}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,16,2} = p_s \frac{\theta_2}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,16,3} = p_s \frac{\theta_3}{\sum \theta_1 + \theta_2 + \theta_3},$$

$$\Pi_{\theta,16,4} = (1 - p_s), \Pi_{\theta,17,5} = \frac{\theta_4}{\sum \theta_4 + \theta_5}\ \Pi_{\theta,17,6} = \frac{\theta_5}{\sum \theta_4 + \theta_5}$$

$$\Pi_{\theta,18,7} = \frac{\theta_1}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,18,8} = \frac{\theta_2}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,18,9} = \frac{\theta_3}{\sum \theta_1 + \theta_2 + \theta_3}$$

$$\Pi_{\theta,19,16} = p_s \frac{\theta_1}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,19,17} = p_s \frac{\theta_2}{\sum \theta_1 + \theta_2 + \theta_3},\ \Pi_{\theta,19,18} = p_s \frac{\theta_3}{\sum \theta_1 + \theta_2 + \theta_3},$$

$$\Pi_{\theta,19,19} = (1 - p_s),$$

Proposition: The Arthur's value function is a limit point of the learning algorithm for strategies.

Proof: First let us check that our learning model satisfies the assumptions of the theorem that we present in Appendix A.

(F) Let $M_R = 2R + U$ where $U = \max_{c_t \in \{0,1,2\}} u(c_t)$ such that for $|\theta| < R$

$$\sup_\theta \sup_x |f(\theta, x)| \le M_R .$$

(M1) Since the transition probability matrix is 'irreducible' and 'recurrent' we have a unique invariant distribution , Γ_θ for every Π_θ. The θ-dependant solution of the equation

$$\Gamma_\theta = \Pi_\theta \; \Gamma_\theta$$

is,

$$\Gamma_\theta = \left[\frac{\theta_1}{\theta_2}, 1, \frac{\theta_3}{\theta_2}, \frac{(1-p_s)(\theta_1+\theta_2+\theta_3)}{p_s\theta_2}, \frac{\theta_4(\theta_1+\theta_2+\theta_3)}{p_s\theta_1(\theta_4+\theta_5)}, \frac{\theta_5(\theta_1+\theta_2+\theta_3)}{p_s\theta_1(\theta_4+\theta_5)}, \frac{\theta_3}{p_s\theta_2}, \frac{\theta_3}{p_s\theta_1}, \right.$$

$$\frac{\theta_3^2}{p_s\theta_1\theta_2}, 1, \frac{\theta_2}{\theta_1}, \frac{\theta_3}{\theta_1}, \frac{(1-p_s)(\theta_1+\theta_2+\theta_3)}{p_s\theta_1}, \frac{\theta_5^2(\theta_1+\theta_2+\theta_3)}{p_s\theta_1\theta_4(\theta_4+\theta_5)}, \frac{\theta_5(\theta_1+\theta_2+\theta_3)}{p_s\theta_1(\theta_4+\theta_5)}, (1-p_s)\frac{(\theta_1+\theta_2)}{p_s\theta_2},$$

$$\left. (1-p_s)\frac{(\theta_1+\theta_2)}{p_s\theta_1}, (1-p_s)\frac{\theta_3(\theta_1+\theta_2)}{p_{s1}\theta_1\theta_2}, \frac{(1-p_s)^2(\theta_1+\theta_2)^2(\theta_1+\theta_2+\theta_3)}{p_s^2\theta_1\theta_2} \right],$$ where the invariant distribution is normalized by the term to make the distribution a probability distribution;

$$Sum = \frac{(\theta_1+\theta_2+\theta_3)(\theta_2\theta_4 + p_s\theta_2\theta_5 + p_s\theta_2\theta_4 + p_s\theta_3\theta_4 + \theta_1\theta_4)}{p_s^2\theta_1\theta_2\theta_4}$$

(M2) For $p = \infty$ and constants $\alpha_R \in (0,1), K_R = 8$, the following inequality holds:

$$\sup_{|\theta|\le R} \int |y|^p \, \Pi_\theta(x;dy) \le \alpha_R |x|^p + K_R \, .$$

(M3) Let $x = Y_i$, then

$$\sup_x \left|\Pi_\theta v(x) - \Pi_{\theta'} v(x)\right| \le \left|\sum_j v(Y_i)\left[\Pi_{\theta,i,j} - \Pi_{\theta',i,j}\right]\right| \le |v(Y_i)| |\theta - \theta'| C_R$$

$$\le K_R |\theta - \theta'| \sup_{x \ne x'} \frac{|v(x) - v(x')|}{|x - x'|}$$

Remark: The operator Π_θ can simply be understood as a matrix operating on R^q via $\Pi_\theta v_i = \sum_j (\Pi_\theta)_{ij} v_j$ where $v_i \equiv v(Y_i)$ for any given function $v : R^k \to R.$ The norm on v is defined by $\sup_{x \neq x'} \frac{|v(x) - v(x')|}{|x - x'|}$

(M4) The solution of the equation

$$(1 - \Pi_\theta) v_\theta = f(\theta,.) - \int f(\theta, y) \Gamma_\theta(dy)$$

is unique and $v_\theta(Y_i) = \sum_i \sum_j a_{ij} (f(\theta, Y_j) - \int f(\theta, Y) \Gamma_\theta(dy)$ satisfies the equation above. (The coefficients a_{ij} could not be written here because of its longness)

(M5) For the above values $v_\theta(Y_i)$, from **(F)** and **(M3)** the following 3 conditions are trivially satisfied.

a) $\sup_{|\theta| \leq R} |v_\theta(x) - v_\theta(x')| \leq M_R |x - x'|$,

b) $\sup_{|\theta| \leq R} |v_\theta(x)| \leq C_R (1 + |x|)$,

c) $|v_\theta(x) - v_{\theta'}(x)| \leq C_R |\theta - \theta'| (1 + |x|)$ for $|\theta| \leq R, |\theta'| \leq R$.

Now we are ready to find the limiting values of strengths. By the Theorem, that we mentioned in the Appendix A , for every θ^* that is a locally asymptotically stable point of the equation

$$\frac{d\theta(t)}{dt} = -\phi(\theta(t))$$

with domain of attraction $D(\theta^*)$ and for every $\omega \in \tilde{\Omega}_1$ such that for some compact $A \subset D(\theta^*), \theta_t(\omega) \in A$ for infinitely many t, the following holds:

$$\lim_t \theta_t(\omega) = \theta^*$$

What we need to find is the solution of equation below:

$$\phi(\theta) \equiv \int f(\theta, y)\Gamma_\theta(dy) = E_{\Gamma_\theta}\left[f(\theta, y)\right] = 0$$

and the required solution of the above equation is the simultaneous solution of the 6 equations written below:

$$l1 := (\theta_1 - u(2) - \beta\theta_1)p_s\theta_1 + (\theta_1 - u(2) - \beta\theta_2)p_s\theta_2 + (\theta_1 - u(2) - \beta\theta_3)p_s\theta_3 + (\theta_1 - u(2) - \beta\theta_6)(1 - p_s)(\theta_1 + \theta_2 + \theta_3),$$

$$l2 := (\theta_2 - u(1) - \beta\theta_4)\theta_4 + (\theta_2 - u(1) - \beta\theta_5)\theta_5,$$

$$l3 := (\theta_3 - \beta\theta_1)\theta_1 + (\theta_3 - \beta\theta_2)\theta_2 + (\theta_3 - \beta\theta)\theta_3,$$

$$l4 := (\theta_4 - u(1) - \beta\theta_1)p_s\theta_1 + (\theta_{41} - u(1) - \beta\theta_2)p_s\theta_2 + (\theta_4 - u(1) - \beta\theta_3)p_s\theta_3 + (\theta_4 - u(1) - \beta\theta_6)(1 - p_s)(\theta_1 + \theta_2 + \theta_3),$$

$$l5 := (\theta_5 - \beta\theta_5)\theta_5 + (\theta_5 - \beta\theta_4)\theta_4,,$$

$$l6 := (\theta_6 - \beta\theta_6)p_s\theta_1 + (\theta_6 - \beta\theta_2)p_s\theta_2 + (\theta_6 - \beta\theta_3)p_s\theta_3 + (\theta_6 - \beta\theta_6)(1 - p_s)(\theta_1 + \theta_2 + \theta_3)$$

When we directly substitute the values of *Arthur's augmented value function* for θ_i in the equations, we observe that these values are in the solution set of $\{l1 = 0, l2 = 0, l3 = 0, l4 = 0, l5 = 0, l6 = 0\}$.

Chapter 4: Simulation Results

To illustrate the operation of our model and to understand the convergence behavior of given initial *strengths*, we prepared a GAUSS program (see Appendix B) to implement the learning algorithm described in Chapter 3. Although, the equivalence of the asymptotically stable points of the Equation (15) and the *Arthur's augmented values* was shown in the Chapter 3, the speed of convergence and the numerical analysis is also of importance.

In the program, the *cooling sequence*, γ_{t+1} is defined as:

$$\gamma_{t+1} = \frac{1}{\tau_{k_t, c_t} + 2}$$

where τ_{k_t, c_t} is an *experience counter*, recording the number of times that the particular classifier (k_t, c_t) has been selected up to time *t*. We set τ_{k_t, c_t} =0, for *t=0* ,so the initial τ_{k_t, c_t} are 0 for all (k_t, c_t). In order to control the speed of convergence of γ_{t+1}, we use a constant *l* in such a way that;

$$\gamma_{t+1} = \frac{1}{\tau_{k_t, c_t} . l + 2}.$$

From now on, for all the numerical analysis, we will fix *l=10* and restrict the number of periods to 2000.

A single run of the program with given initial strengths $S_{22} = 10$, $S_{21} = 10$, $S_{20} = 10$, $S_{11} = 10$, $S_{10} = 100$, $S_{00} = 10$ and $p_s = 0.2$ is seen in Figure1. The given initial strengths are not consistent with that of Arthur's augmented values both in ordering and size . But even for this case, after 2000 runs, the associated strengths S_{11}, S_{10} fall into a neighborhood , with radius 2 of $\bar{v}_r(1,1)$, $\bar{v}_r(1,0)$ respectively .

To check the pattern of S_{22}, S_{21}, S_{20}, we have set the initial values as $S_{22}=100,\ S_{21}=10,\ S_{20}=100,\ S_{11}=10,\ S_{10}=10,\ S_{00}=10$ and $p_s=0.2$. As it can be seen from the Figure 2, after 2000 run, the strengths S_{22}, S_{21}, S_{20} are respectively, 24.92, 29.52, 26.27. Again, we can observe that the strengths converges to their target values independent from their inital values. In this case the values are closer to *Arthur's augmented values*.

We have taken $S_{22}=10,\ S_{21}=10,\ S_{20}=10,\ S_{11}=10,\ S_{10}=100,\ S_{00}=10$,as the initial values of strengths and the probilities $p_s=0.4$ in Figure3, $p_s=0.6$ in Figure5 .In both Figure3 and Figure5, the observations that we have mentioned for Figure1 is valid.

In Figure 4 and 6 we showed the convergence behavior of S_{22}, S_{21} and S_{00} with given inital values as: $S_{22}=100,\ S_{21}=10,\ S_{20}=100,\ S_{11}=10,\ S_{10}=10,\ S_{00}=10$ and the probability values of $p_s=0.4$, $p_s=0.6$ respectively.

The *cooling sequence* here has a great importance. To show the effect of it, we run the program for the case *l*=1, and the rest is the same of Figure 4. In this case the learning process is slow, as can be seen from the Figure 7. However, the fluctuations, which happens for greater *l* values does not happen in this case.

Chapter 5: Concluding Remarks

In this dissertation, we suggested an alternative behavioral model to explain the excess sensitivity of consumption to temporary income shocks. The model incorporates classifier systems learning and a stochastic decisions where likelihood depends on the relative strengths of their perceived payoffs. We show the convergence of these perceived payoffs and characterize their limit points. The limit points can independently be calculated using a functional equation analogous to Bellman's equation.

When we applied this algorithm to a cake-eating problem, we observed that overconsumption is more likely than underconsumption.

Our approach is an alternative to the rule-based decision theory (studied by Lettau and Uhlig, 1999, in a similar setup) in the spirit of case based decision theory of Gilboa and Schmeidler (1995).

References

Arthur, Brian W.,1989.”Dynamics of Classifier Competitions”. Mimeo, Santa Fe Institute.

Binmore, Kenneth G. and Samuelson, Larry.,1992. “Evolutionary Stability in Repeated Games Played by Finite Automata.” *Journal of Economic Theory*, 57(2), pp. 278-305.

Campbell, John Y. and Mankiw, N. Gregory.,1990. “Permanent Income, Current Income, `and Consumption.” *Journal of Business and Economic Statistics, 8(3),* pp. 265-79.

Gilboa, Yitzhak and Schmeidler, David, 1995. “Case Based Decision Theory”. *Quarterly Journal of Economics*, 110. pp.605-639.

Flavin, Marjorie A.,1981. “The Adjustment of Consumption to Changing Expectations about Future Income.” *Journal of Political Economy*, 89(5), pp.974-1009.

Friedman, Milton., 1957. *A theory of the consumption function*. Princeton, NJ: Princeton University Press.

Holland, John H. *et al*., 1975.”Adaptation in natural and Artificial Systems” University of Michigan Press, Ann Arbor.

Ingram, Beth Fisher.,1990. “Equilibrium Modelling of Asset Prices: Rationality versus Rules of Thumb”. *Journal of Business and Economic Statistics*, 8(1), pp:115-25.

Lettau, Martin. and Uhlig, Harald.,1999. “Rules of Thumb versus Dynamic Programming.” *The American Economic Review*,89(1), pp.148-174.

Ljung, Lennart.,1977. “Analysis of Recursive Stochastic Algorithms.” *IEEE Transactions on Automatic Control*, 22(4), pp.551-575.

Ljung, L; Pflug, G; and Walk, H.,1992. *Stochastic approximation and optimization of random systems*. Basel, Switzerland: Birkhauser-Verlag.

Lusardi, Annamaria.,1996. “Permanent Income, Current Income, and Consumption: Evidence From two Panel Data Sets.” *Journal of Business and Economics Statistics*, 14(1), pp.81-90.

Metivier,, Michel. And Priouret, Pierre. 1984. “Applications of a Kushner and Clark Lemma to General Classes of Stochastic Algorithms.” *IEEE Transactions on Information Theory*, 30(2), pp. 140-151

Rust, John., 1992. “Do People Behave According to Bellman’s Principle of Optimality?” Hoover Institute Working Papers in Economics, E-92-10..

Sargent, Thomas J., 1993. *Bounded rationality in macroeconomics*. Arne Ryde memorial lectures. Oxford: Oxford University Press, 1993.

Simon, Herbert.,1982. *Models of bounded rationality*. Cambridge, MA: MIT Press.

Stokey, N. L., Lucas, R. Jr., with Prescott, E. C.,1989. *Recursive methods in Economic Dynamics* . Harvard University Press, Cambridge.

Zeldes, Stephen P., 1989. “Consumption and Liquidity Constraints: An Empirical Investigation.” *Journal of Political Economy*, 97(2), pp.305-406

Appendix A

A theorem about Markov Stochastic Approximation Algorithms.

In this section, we use the notation of Metivier and Priouret (1984). For a general overview and introduction to stochastic approximation algorithms, see Sargent (1992) and Ljung, Pflug and Walk (1992).

For each $\theta \in R^d$ consider a transition probability $\Pi_\theta(y;dx)$ on R^k. This transition probability defines a controlled Markov chain on R^d.

Define a stochastic algorithm by the following equations:

$\theta_{t+1} = \theta_t - \gamma_{t+1} f(\theta_t, Y_{t+1})$ (14) where $f : R^d \times R^k \to R^d$

Call $P[Y_{t+1} \in B | \delta_t] = \Pi_{\theta_t}(Y_t, B)$ where $P[Y_{t+1} \in B | \delta_t]$ is the conditional probability of the event $Y_{t+1} \in B$ given $\theta_0, ..., \theta_t, Y_0, ..., Y_t$.

We call $\Psi \to \Pi_\theta \Psi$ the operator $\Pi_\theta \Psi(X) \equiv \int \Psi(y) \Pi_\theta(x;dy)$. Assume the following:

F) For every R>0 there exists a constant M_R such that for $|\theta| < R$

$$\sup_\theta \sup_x |f(\theta, x)| \le M_R .$$

M1) For every θ, the Markov chain Π_θ has a unique invariant probability Γ_θ.

M2) There exists $p \ge 2$ and positive constants $\alpha_R < 1, K_R$ for which

$$\sup_{|\theta|\le R} \int |y|^p \Pi_\theta(x;dy) \le \alpha_R |x|^p + K_R .$$

M3) For every function v with the property $|v(x)| \le K(1+|x|)$ and every $\theta, \theta^{'}$ with the property $|\theta| \le R, |\theta^{'}| \le R$,

$$\sup_x |\Pi_\theta v(x) - \Pi_{\theta^{'}} v(x)| \le \tilde{K}_R |\theta - \theta^{'}| \sup_{x \ne x^{'}} \frac{|v(x) - v(x^{'})|}{|x - x^{'}|}$$

M4) For every θ the Poisson equation

$$(1 - \Pi_\theta) v_\theta = f(\theta,.) - \int f(\theta, y) \Gamma_\theta(dy)$$

has a solution v_θ with the following properties of **M5**;

M5) For all R there exists constants M_R, C_R so that

a) $\sup_{|\theta|\le R} |v_\theta(x) - v_\theta(x^{'})| \le M_R |x - x^{'}|$**,**

b) $\sup_{|\theta|\le R} |v_\theta(x)| \le C_R(1+|x|)$,

c) $|v_\theta(x) - v_{\theta^{'}}(x)| \le C_R |\theta - \theta^{'}| (1+|x|)$ for $|\theta| \le R, |\theta^{'}| \le R$

Let

$$\phi(\theta) \equiv \int f(\theta, y) \Gamma_\theta(dy) = E_{\Gamma_\theta}[f(\theta, y)]$$

Metivier and Priouret (1984) have shown the following theorem.

Theorem: Consider the algorithm defined above and assume that **(F)** and **(M1)** through **(M5)** satisfied. Suppose that (γ_t) is decreasing with $\sum_t \gamma_t = +\infty$ and $\sum_t \gamma_t^{1+\left(p/2\right)} < \infty$, where $p \ge 2$ is the constant entering **(M2)**. Let $\Omega_1 \equiv \{\sup_t |\theta_t| < \infty\}$. Then there is a set

$\tilde{\Omega}_1 \subset \Omega$ such that $P(\Omega \setminus \Omega_1) = 0$ and with the following property: for every θ^* that is a locally asymptotically stable point of the equation

$$\frac{d\theta(t)}{dt} = -\phi(\theta(t))$$

with domain of attraction $D(\theta^*)$ and for every $\omega \in \tilde{\Omega}_1$ such that for some compact $A \subset D(\theta^*), \theta_t(\omega) \in A$ for infinitely many *t*, the following holds:

$$\lim_t \theta_t(\omega) = \theta^*$$

APPENDIX B

```
/*
Author:        Erdem Basci and Mahmut Erdem
Last revision: 31 January 2001

This program simulates classifier systems learning
for agents facing dynamic programming problems.
of the type Basci and Erdem (2001).

Let the matrix U(k,m) denote the utilities from state-action
pairs (k,m).

Let KK={1,...,k} be the state space and
MM={1,...,m} denote the action space.

Let G(k,m) denote the feasibility matrix, with entries=1 denoting
m is feasible at k and 0 denoting it to be infeasible.

Let KP(k,m) denote the pre-shock transition matrix, giving the
next period's state kp as a function of (k,m).

Let KN(kp,kn) denote the probability distribution matrix giving
the probability of next period's state, knext, being less than or equal
to kn, given the pre-shock state, k_p.

Let S(k,m) denote the strength matrix, for classifiers.

Let T(k,m) denote the number of times that classifier (k,m) have
been activated in the past (experience counter matrix).


k:  current state
kprev:  previous period's state
knext:  next period's state

m:  current consumption
mprev:  previous period's consumption
mnext:  next period's consumption


Let D(k,m) denote the action density matrix and
let DC(k,m) denote the cummulative action distribution matrix.

Let pr denote the probability of random action.

Let l denote the inverse of cooling speed parameter.

===============INITIALIZATION==============================*/

beta=0.9;

KK={1,2,3};   /* (0,1,2) units of cake respectively  */
MM={1,2,3};   /* (0,1,2) units of consumption respectively */

U={0 0 0,
   0 8 0,
   0 8 10};

G={1 0 0,
```

```
   1 1 0,
   1 1 1};

KP={1 0 0,
    2 1 0,
    3 2 1};

KN={0.6 0.6   1,    /* State (cummul.) distr. due to subsidy shock */
      0  1    1,
      0  0    1};

D=zeros(3,3);      /* Action density at each state */
DC=zeros(3,3);     /* Action (cummulative) distribution at each state */

l=10;

k=3;   /* Two units of cake to start with  */

kprev=1;
mprev=1;

T=zeros(3,3);            /*  Experience counters start at zero  */

/* S=(20*rndn(3,3)+46).*G;  Infeasible ones set to zero  */


S={10.23      0        0,
   10.41   10.23       0,
   10.27   10.41      10.00};

/* S=rndu(3,3)+1;    Negative values not allowed  */

S=S.*G;    /* Only feasible ones are positive  */

SHIST=Zeros(2001,4);

SHIST[1,.]=k~S[3,.];

n=1;              /* period counter */

do while n<=2000;

/* ============MAIN ALGORITHM======================*/


/* ----------Action determination-----------*/

gcount=sumc(G[k,.]');        /* number of feasible actions at k */

si=1;
do while si<=3;
   D[si,.]=S[si,.]/sumc(S[si,.]');   /* fill action densities */
   r=1;
   do while r<=3;
     DC[si,r]=sumc(D[si,1:r]');  /* generate action cummulative probabilities */
     r=r+1;
   endo;
   si=si+1;
endo;

shock1=rndu(1,1);
```

```
i=1;
do while shock1>DC[k,i];
   i=i+1;
endo;

m=i;   /* chosen action  */


/* ---------Next period's state -------------*/

kpre=KP[k,m];  /* pre-shock state determined by state k and action m */

shock2=rndu(1,1);
i=1;
do while shock2>KN[kpre,i];
   i=i+1;
endo;

knext=i;  /* post-shock state is determined by the conditional pdf matrix KN */


/* ---------Strength update ----------------*/


if n>=2;

gcountn=sumc(G[knext,.]');        /* number of feasible actions at knext */

S[kprev,mprev]=S[kprev,mprev]+1/(T[kprev,mprev]/l+2)*(U[kprev,mprev]+beta*
          S[k,m]- S[kprev,mprev]);

T[kprev,mprev]=T[kprev,mprev]+1;

endif;

/* ==========END OF MAIN ALGORITHM============== */


kprev=k;
mprev=m;
k=knext;

SHIST[n+1,.]=k~S[3,.];

n=n+1;

endo;

/* output file=A:\lartcl.out; */

output reset;
SHIST;
output off;
```

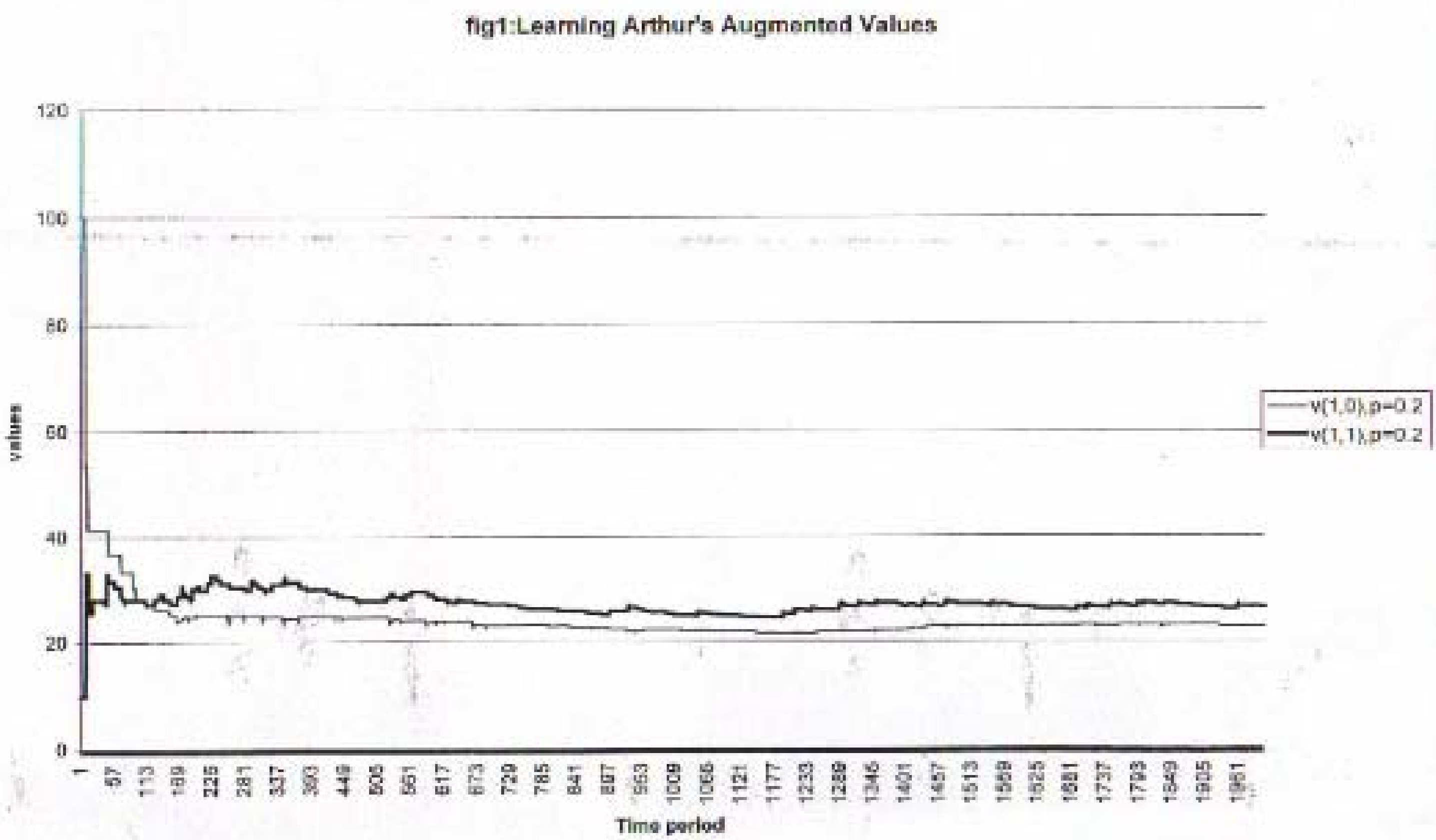
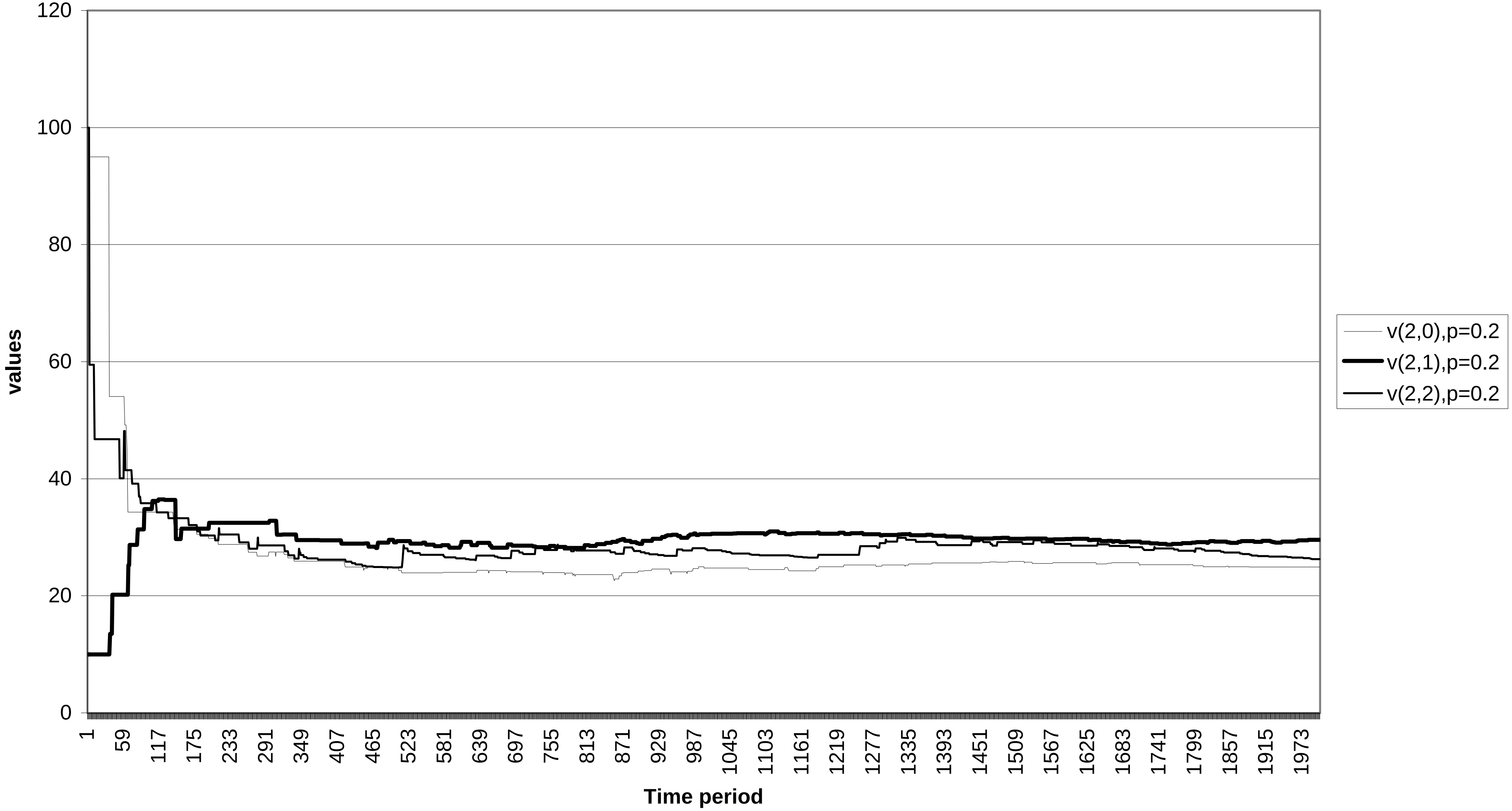


fig1:Learning Arthur's Augmented Values
values
Time period
v(1,0),p=0.2
v(1,1),p=0.2
0
20
40
60
80
100
120

fig2:Learning Arthur's Augmented Values



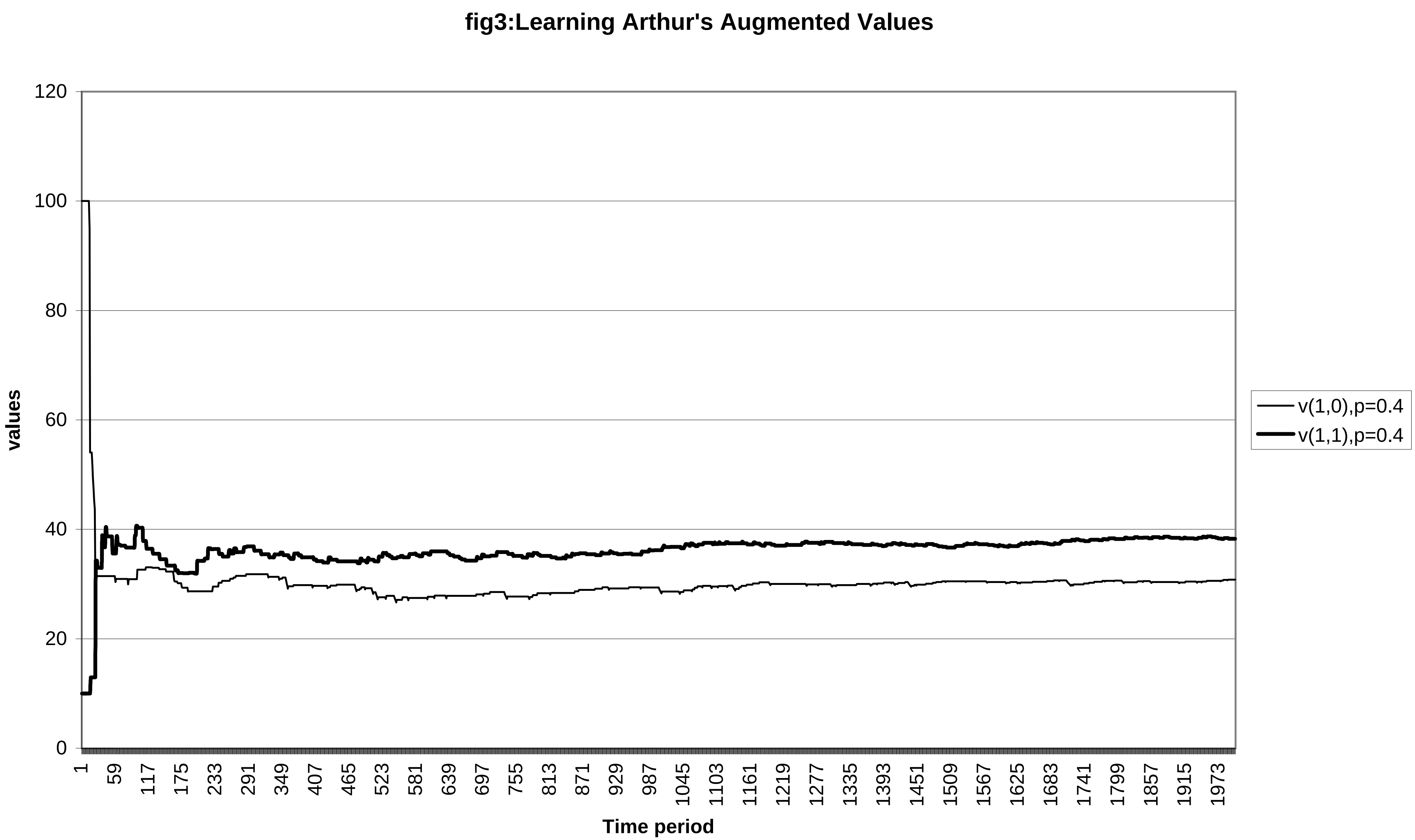


fig3:Learning Arthur's Augmented Values
values
120
100
80
60
40
20
0
1
59
117
175
233
291
349
407
465
523
581
639
697
755
813
871
929
987
1045
1103
1161
1219
1277
1335
1393
1451
1509
1567
1625
1683
1741
1799
1857
1915
1973
Time period
v(1,0),p=0.4
v(1,1),p=0.4

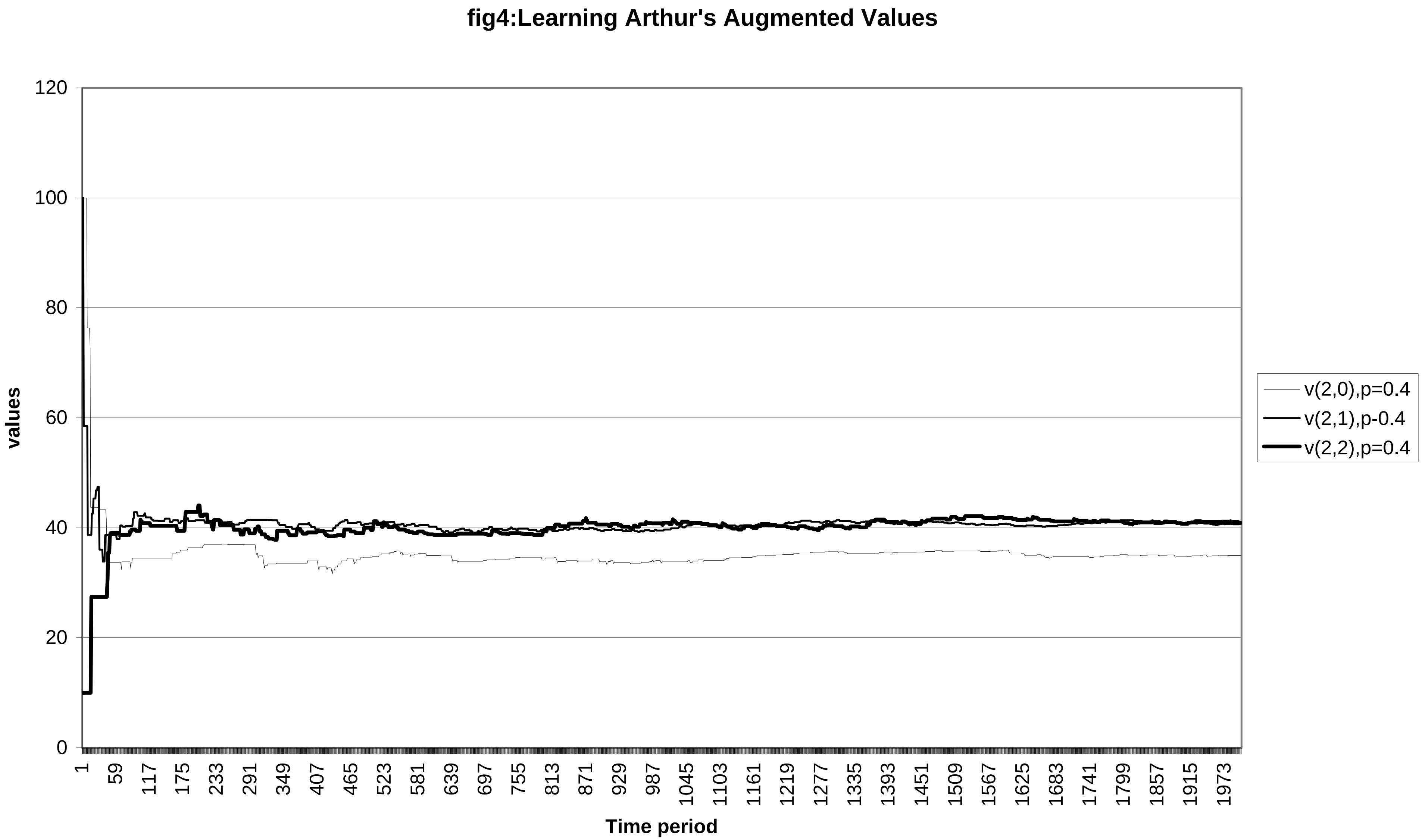
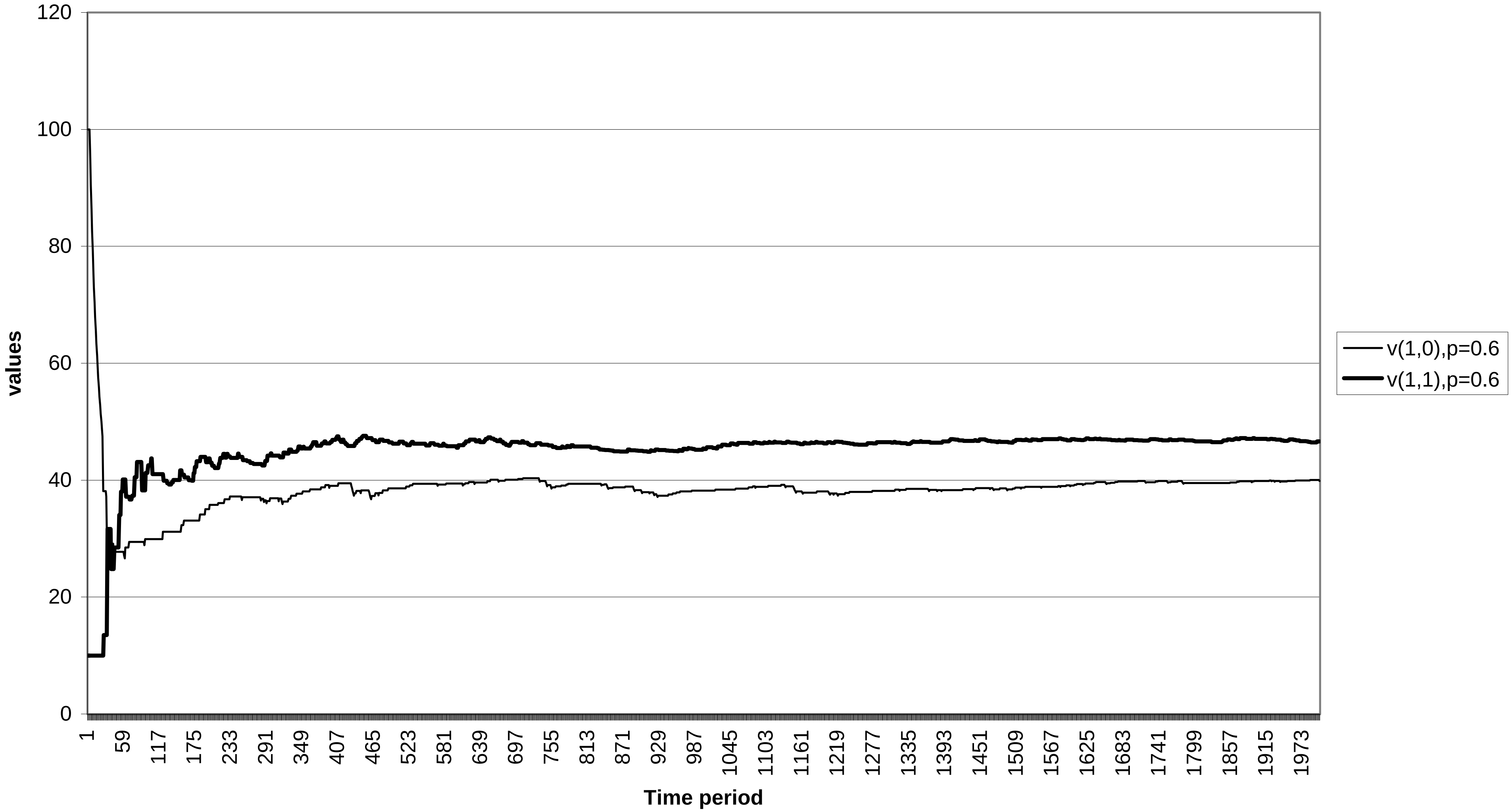


fig4:Learning Arthur's Augmented Values
values
120
100
80
60
40
20
0
1
59
117
175
233
291
349
407
465
523
581
639
697
755
813
871
929
987
1045
1103
1161
1219
1277
1335
1393
1451
1509
1567
1625
1683
1741
1799
1857
1915
1973
Time period
v(2,0),p=0.4
v(2,1),p-0.4
v(2,2),p=0.4

fig5:Learning Arthur's Augmented Values



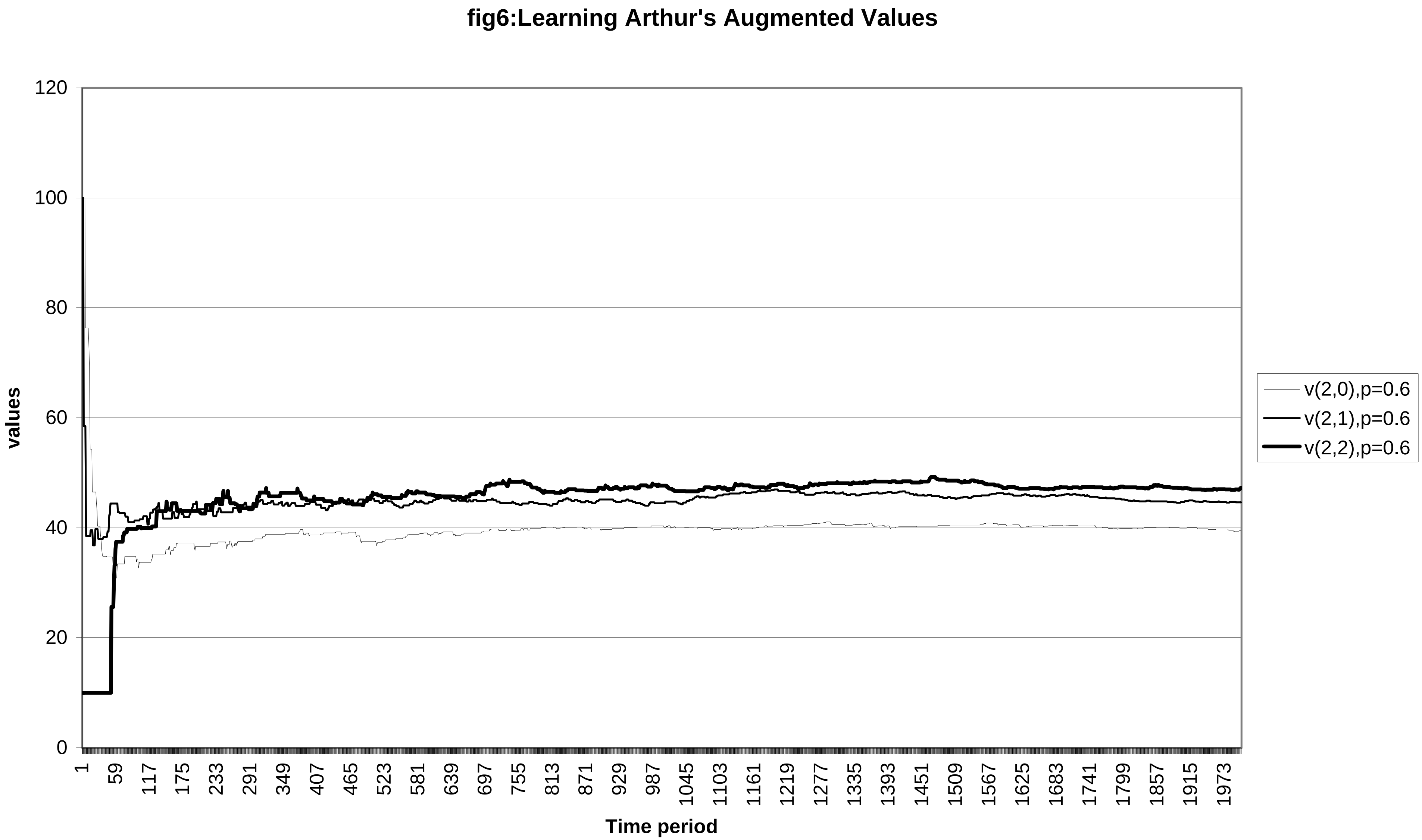


fig6:Learning Arthur's Augmented Values
values
Time period
v(2,0),p=0.6
v(2,1),p=0.6
v(2,2),p=0.6
0
20
40
60
80
100
120
1
59
117
175
233
291
349
407
465
523
581
639
697
755
813
871
929
987
1045
1103
1161
1219
1277
1335
1393
1451
1509
1567
1625
1683
1741
1799
1857
1915
1973

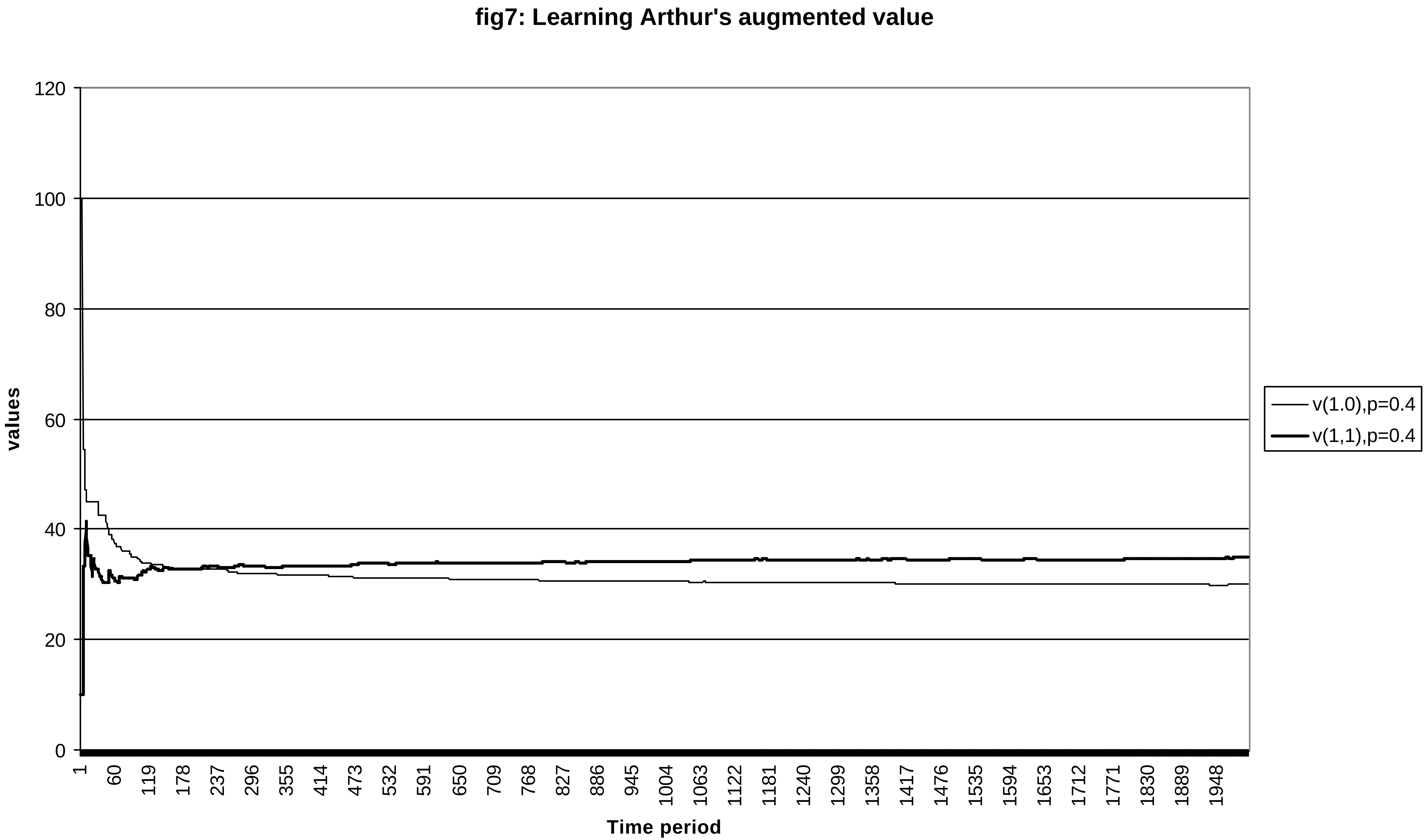


fig7: Learning Arthur's augmented value
values
0
20
40
60
80
100
120
1
60
119
178
237
296
355
414
473
532
591
650
709
768
827
886
945
1004
1063
1122
1181
1240
1299
1358
1417
1476
1535
1594
1653
1712
1771
1830
1889
1948
Time period
v(1.0),p=0.4
v(1,1),p=0.4